FMKit - An In-Air-Handwriting Analysis Library and Data Repository

Duo Lu, Linzhen Luo, Dijiang Huang, Yezhou Yang
Arizona State University
{duolu, lluo21, dijiang.huang, yz.yang}@asu.edu

Abstract

Hand-gesture and in-air-handwriting provide ways for users to input information in Augmented Reality (AR) and Virtual Reality (VR) applications where a physical keyboard or a touch screen is unavailable. However, understanding the movement of hands and fingers is challenging, which requires a large amount of data and data-driven models. In this paper, we propose an open research infrastructure named FMKit for in-air-handwriting analysis, which contains a set of Python libraries and a data repository collected from over 180 users with two different types of motion capture sensors. We also present three research tasks enabled by FMKit, including in-air-handwriting based user authentication, user identification, and word recognition, and preliminary baseline performance.

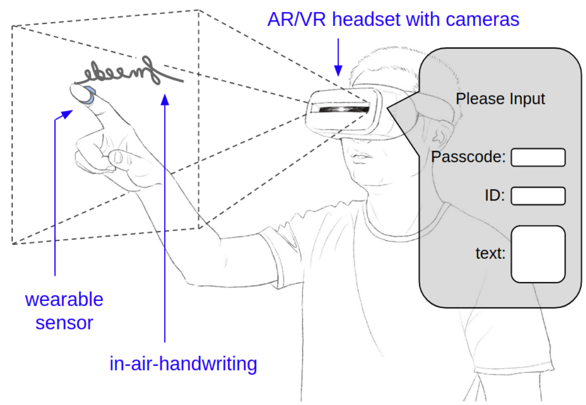

Figure 1. In-air-handwriting as input to AR/VR applications.

1. Introduction

In AR/VR applications, 3D in-air hand gestures are natural ways to interact with virtual objects as well as a method to input information [3, 9, 10, 14]. Besides simple gestures like tap, scroll, slide, etc, there are cases when complicated information such as text or identity is needed to be presented to the software as input. Consider the scenario shown in Figure 1, a dialog pops up to an AR/VR user asking a passcode, an ID, or a short piece of text information (*e.g.*, a tweet) as input information. One possible method is using in-air-handwriting [4, 15], *i.e.*, instead of typing, the user can write a piece of information in the air, and ask the computer to recognize it. However, there are a few challenges. First, recognizing patterns in hand motion generally requires a large amount of data to train machine learning models, but the data collection is generally expensive. Second, there is no standard hand motion capture device and data collection procedures, and existing works choose their own ways, which makes the datasets incompatible and hard to compare. Additionally, most of them are not openly available. Third, given different motion capture devices and multiple in-air-handwriting related tasks solved using various algorithms, these mix-and-matches make it difficult to fairly evaluate different systems and compare their performance results [2]. To the best of our knowledge, the only open datasets of in-air-handwriting are constructed for word recognition [11, 4] using the Leap Motion controller.

In this paper, we propose FMKit [1] (Finger Motion analysis Kit), an open-source library and open datasets for in-air-handwriting analysis [2]. Currently, FMKit contains four datasets of in-air-handwriting motion signals, in total 103K data signals. Especially, each dataset has two parts collected in identical scenarios with the same in-air-handwriting content using two different motion capture devices, so that the results of the two devices can be compared. We also release the details of the devices, data collection protocols, and the source code of the client software online [3] so that researchers can extend our datasets. Meanwhile, we propose three types of research tasks enabled by the FMKit infrastructure: (a) user authentication [1, 12, 13, 6, 8], (b) user identification [5, 7], and (c) in-air-handwriting word recognition [4, 15], which corresponds to the three types of inputs in Figure 1. Additionally, we provide the baseline performance for these tasks with our dataset. We hope this open infrastructure helps researchers to validate models, bench-

[1]This project is supported by NSF (award #1925709).
[2]Video Demo: `https://youtu.be/O3Jqq9yqJSE`
[3]https://github.com/duolu/fmkit

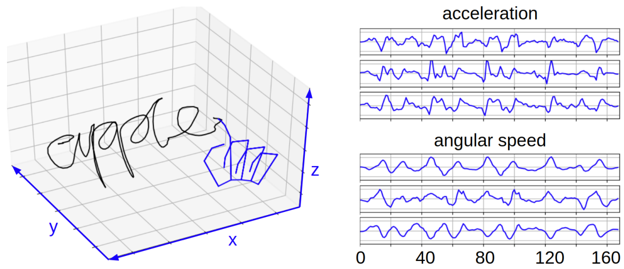

Figure 2. An example of the finger motion trajectory of writing "apple" together with the hand skeleton captured by the Leap Motion controller (left), and the signal containing the acceleration and the angular speed obtained from the data glove (right).

mark the performance of algorithms, and further facilitate data-driven approaches that can be generalized across sensor types for in-air-handwriting analysis.

2. The FMKit Library and Data Repository

In the FMKit data repository, two different types of devices are used to capture the hand movement: the Leap Motion controller and a custom made data glove with an Inertial Measurement Unit (IMU) at the tip of the index finger. The Leap Motion controller can capture the 3D position of each joint of the hand at about 110 Hz (usually only the tip of the index finger or the palm center is interested). The data glove can track the 3D acceleration and angular speed of the tip of the index finger at 50 Hz. The basic unit of data in our datasets is called a "signal", which is a recorded piece of hand movement of writing a string in the air. An example of a signal is shown in Figure 2. A Python module is constructed to obtain other physical states such as the orientation, and conduct other preprocessing such as resample, posture normalization, filter, etc. Additionally, a Python-based data browser is also built by us to facilitate the inspection and visualization of signals.

The following datasets are provided.

(1) ID-passcode: We asked 180 participants to create 360 distinct meaningful strings as ID or passcode and write them in-the-air (exactly two strings for each person). Such a string can contain alphanumeric letters, characters in a language other than English, or symbols like five-pointed stars which cannot be directly typed. For each string, the person wrote five repetitions as registration and another five repetitions as login attempts, using both devices, which resembles the sign-up and sign-in procedure. There is no restriction on the persons' writing behavior, for example, one may write a string in a legible way or write it fast in an illegible way like a signature. In total, there are 7,200 signals.

(2) ID-passcode-collision: We asked 10 imposters to write the strings created by the users in the first dataset. In this setting, the imposters know the meaning of the strings, but the imposters have not seen the original participants writing them in the air, which resembles the case of ID or passcode collision. For each string in the first dataset, one imposter wrote it with five repetitions with each type of device. In total, there are 36,000 signals.

(3) ID-passcode-spoofing: Similar to the second dataset, we asked 10 imposters to imitate the in-air-handwriting of the strings generated by the users in the first dataset. In this setting, the imposters can watch the recorded video of the in-air-handwriting and they were also informed with the meaning of the in-air-handwriting. 180 strings in the first dataset are spoofed and each imposter wrote the string with five repetitions using both devices. In total, there are 18,000 signals.

(4) word-210: We asked 10 participants to write 210 English words and 210 Chinese words. For each word, each person wrote it with five repetitions using both devices. In total, there are 42,000 signals.

Besides, we are currently working on a superset of the word-210 dataset, named **word-10k**. As the name suggests, this dataset contains 10,000 English words and 10,000 Chinese words, where each English word is written by 12 different users with one repetition and each Chinese word is written by 18 different users with one repetition. In total, there will be 600,000 signals.

3. Tasks and Experiments with FMKit

A signal in any of the five datasets can be represented as a tuple $(\mathbf{x}, y)$, where $\mathbf{x}$ is the signal data, and y is the label indicating the meaning of the string. For example, in the first dataset, there are 360 different labels. Based on this definition, we propose the following three types of tasks.

3.1. User Authentication

User authentication is essentially a binary classification problem, which serves as the function of "typing a password" over the gesture interface. In this task, each of the 360 strings in the first dataset is considered as the passcode of an account. The five signals for registration are used to construct templates and train discriminative models, and the five signals for login requests are used to test the performance. Specifically, consider an account constructed by the signals with label y and another login signal $(\mathbf{x}, y')$ in the first dataset, the authentication system should accept the signal if $y = y'$ (*i.e.*, it is generated by the same person writing the same string), and reject it otherwise. We call them the positive testing signals and the negative testing signals respectively. The signals in the second and third datasets are used to evaluate the performance in scenarios with adversaries, *i.e.*, all of them are negative testing signals that should be rejected. Due to the imbalance of the number of positive testing signals (only five) and the number of negative testing signals (potentially thousands or more) for one account, evaluation metrics used commonly in biometrics systems such as False Accept Rate (FAR) and False Reject Rate (FRR) are used instead of precision and recall. Usually, the authentication system has a configurable discriminative threshold that can be changed to trade off FAR and FRR. For example, if the threshold is smaller, there will be more false rejects but less false accepts. When FAR is equal to FRR, this rate is called the Equal Error Rate (EER). When the threshold is set to a value such that the FAR is 10^{-3} or 10^{-4}, the corresponding FRR is called FAR1K or FAR10K.

We set up three different experiments for each account: (a) all testing signals are from the first dataset, (b) all positive testing signals are from the first dataset but all negative testing signals are from the second dataset, (c) all positive testing signals are from the first dataset but all negative testing signals are from the third dataset. The performance results are averaged over all accounts and baselines are provided in Table 1. They are obtained using a per-account SVM classifier on aligned and temporally normalized signals. Details will be posted online together with the dataset.

3.2. User Identification

User identification is essentially a classification problem, which serves as the function of "typing an account ID". In this task, each of the 360 strings in the first dataset is considered as an account ID instead of a passcode. the five signals for registration are used to train a classifier, and the five signals for login requests are used to test the performance. Given a signal $\mathbf{x}$, the classifier determines its label y which represents the corresponding account. Different from a passcode, an ID is usually not a secret. Hence, this task is more like retrieving the top-k candidate accounts using a signal. Together with the user authentication function, a login system can take both the in-air-handwriting of an ID and a passcode to first narrow down to one or a few candidate accounts using the ID and then check each candidate account using the passcode to authenticate the user. In this case, the ID is not needed to be returned to the user. For closed-set identification, the system will always locate at least one account, while for open-set identification, the system may end up with no candidate account and reject the in-air-handwriting ID. The performance metrics are the top-1 and top-5 identification accuracy and the baselines are provided in Table 2. They are obtained using a convolutional neural network encoder trained with pairwise loss, and the identification is done by top-k nearest neighbor search in the latent space. Details will be posted online together with the dataset.

3.3. In-Air-Handwriting Word Recognition

In-air-handwriting word recognition is essentially another classification problem, but the label of each signal is from a word lexicon of the "word-210" dataset instead of a set of user accounts in user identification. Meanwhile, in this task, behaviors variations from different people writing the same word need to be tolerated. For example, one person may write the same letter "O" clockwise but another person may write it counterclockwise. Also, variation of stroke sequences need to considered. For example, when writing the same letter "t", one person may write the horizontal stroke as the last stroke but another person may write it as the first stroke. These problems are challenging especially on our dataset with a limited number of writers. The performance metrics are the top-1 and top-5 classification accuracy and the baselines are provided in Table 3. They are obtained using a similar convolutional neural network as in the previous task. Details will be posted online together with the dataset. Current results are on a dataset with a small lexicon and we believe it would be more challenging with a large lexicon.

	Leap Motion			data glove		
	EER	FAR 1K	FAR 10K	EER	FAR 1K	FAR 10K
(a)	0.12%	0.17%	0.78%	0.17%	0.22%	0.83%
(b)	3.18%	12.9%	25.4%	1.33%	4.72%	6.89%
(c)	3.92%	25.4%	38.2%	1.89%	9.78%	37.1%

Table 1. Baseline performance for user authentication.

	Leap Motion		data glove	
	top-1 accuracy	top-5 accuracy	top-1 accuracy	top-5 accuracy
open-set	93.2%	94.5%	95.7%	96.7%
closed-set	96.7%	98.9%	97.9%	98.8%

Table 2. Baseline performance for user identification.

	Leap Motion		data glove	
	top-1 accuracy	top-5 accuracy	top-1 accuracy	top-5 accuracy
English	79.7%	93.2%	78.5%	93.8%
Chinese	87.4%	96.8%	83.4%	94.4%

Table 3. Baseline performance for in-air-handwriting word recognition, with the "word-210" dataset.

4. Conclusions

In this paper, we provide FMKit, an open-source library and data repository for analysis of in-air-handwriting hand motion signals, which can potentially benefit AR/VR applications. We also propose three research tasks that can be enabled by FMKit, with preliminary baseline performance results. The construction of FMKit is work-in-progress and we hope it can help other research work in this area and pave the road to practical usage of in-air-handwriting as an input method.

References

[1] Gonzalo Bailador, Carmen Sanchez-Avila, Javier Guerra-Casanova, and Alberto de Santos Sierra. Analysis of pattern recognition techniques for in-air signature biometrics. *Pattern Recognition*, 2011.

[2] Gradeigh D Clark and Janne Lindqvist. Engineering gesture-based authentication systems. *IEEE PerCom*.

[3] Euan Freeman, Stephen Brewster, and Vuokko Lantz. Do that, there: an interaction technique for addressing in-air gesture systems. In *CHI*, 2016.

[4] Ji Gan and Weiqiang Wang. In-air handwritten english word recognition using attention recurrent translator. *Neural Computing and Applications*, pages 1–18, 2017.

[5] Eiji Hayashi, Manuel Maas, and Jason I Hong. Wave to me: user identification using body lengths and natural gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3453–3462, 2014.

[6] Duo Lu, Dijiang Huang, Yuli Deng, and Adel Alshamrani. Multifactor user authentication with in-air-handwriting and hand geometry. In *2018 International Conference on Biometrics (ICB)*. IEEE, 2018.

[7] Duo Lu, Dijiang Huang, and Anshul Rai. Fmhash: Deep hashing of in-air-handwriting for user identification. 2019.

[8] Duo Lu, Kai Xu, and Dijiang Huang. A data driven in-air-handwriting biometric authentication system. In *IJCB*. IEEE, 2017.

[9] Keenan R May, Thomas M Gable, and Bruce N Walker. Designing an in-vehicle air gesture set using elicitation methods. In *Proceedings of the 9th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, 2017.

[10] Thammathip Piumsomboon, Adrian Clark, Mark Billinghurst, and Andy Cockburn. User-defined gestures for augmented reality. In *IFIP Conference on Human-Computer Interaction*. Springer, 2013.

[11] Xiwen Qu, Weiqiang Wang, and Ke Lu. In-air handwritten chinese character recognition using discriminative projection based on locality-sensitive sparse representation. In *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pages 1137–1140. IEEE, 2016.

[12] Napa Sae-Bae, Jonathan Wu, Nasir Memon, Janusz Konrad, and Prakash Ishwar. Emerging nui-based methods for user authentication: A new taxonomy and survey. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2019.

[13] Jing Tian, Chengzhang Qu, Wenyuan Xu, and Song Wang. Kinwrite: Handwriting-based authentication using kinect. In *NDSS*, 2013.

[14] Koji Tsukada and Michiaki Yasumura. Ubi-finger: Gesture input device for mobile use. In *Ubicomp 2001 Informal Companion Proceedings*.

[15] Ning Xu, Weiqiang Wang, and Xiwen Qu. Recognition of in-air handwritten chinese character based on leap motion controller. In *International conference on image and graphics*, pages 160–168. Springer, 2015.