BlinKey: A Two-Factor User Authentication Method for Virtual Reality Devices

HUADI ZHU, The University of Texas at Arlington
WENQIANG JIN, The University of Texas at Arlington
MINGYAN XIAO, The University of Texas at Arlington
SRINIVASAN MURALI, The University of Texas at Arlington
MING LI, The University of Texas at Arlington

Virtual Reality (VR) has shown promising potentials in many applications, such as e-business, healthcare, and social networking. Rich information regarding user's activities and their online accounts is stored in VR devices. If it is carelessly unattended, then attackers, including insiders, can make use of the stored information to, for example, perform in-app purchases at the legitimate owner's expenses. Current solutions, mostly following schemes designed for general personal devices, have been proved vulnerable to shoulder-surfing attacks due to the sight blocking caused by the headset. Although there have been efforts trying to fill this gap, they either rely on some highly advanced equipment, such as electrodes to read brainwaves, or introduce heavy cognitive load that has users perform a series of cumbersome authentication tasks. Therefore, an authentication method for VR devices that is robust and convenient is in dire need.

In this paper, we present the design, implementation, and evaluation of a two-factor user authentication scheme, *BlinKey*, for VR devices that are equipped with an eye tracker. A user's secret passcode is a set of recorded rhythms when he/she blinks, together with the unique pupil size variation pattern. We call this passcode as a blinkey, which can be jointly characterized by knowledge-based and biometric features. To examine the performances, *BlinKey* is implemented on an HTC Vive Pro with a Pupil Labs eye tracker. Through extensive experimental evaluations with 52 participants, we show that our scheme can achieve the average EER as low as 4.0% with only 6 training samples. Besides, it is robust against various types of attacks. *BlinKey* also exhibits satisfactory usability in terms of login attempts, memorability, and impact of user motions. We also carry out questionnaire-based pre-/post-studies. The survey result indicates that *BlinKey* is well accepted as a user authentication scheme for VR devices.

CCS Concepts: • **Security and privacy** → **Usability in security and privacy**; • **Human-centered computing** → **Ubiquitous and mobile computing systems and tools**.

Additional Key Words and Phrases: Two-factor authentication, blinking rhythm, pupil size variation, VR device

ACM Reference Format:
Huadi Zhu, Wenqiang Jin, Mingyan Xiao, Srinivasan Murali, and Ming Li. 2020. BlinKey: A Two-Factor User Authentication Method for Virtual Reality Devices. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 4, Article 164 (December 2020), 29 pages. https://doi.org/10.1145/3432217

Authors' addresses: Huadi Zhu, The University of Texas at Arlington, huadi.zhu@mavs.uta.edu; Wenqiang Jin, The University of Texas at Arlington, wenqiang.jin@mavs.uta.edu; Mingyan Xiao, The University of Texas at Arlington, mingyan.xiao@mavs.uta.edu; Srinivasan Murali, The University of Texas at Arlington, srinivasan.murali@mavs.uta.edu; Ming Li, The University of Texas at Arlington, ming.li@uta.edu.

2474-9567/2020/12-ART164 $15.00
https://doi.org/10.1145/3432217

1 INTRODUCTION

1.1 Motivation

Virtual Reality (VR) is an immersive technology that allows users to experience a virtual world with a head mount device. The rapid development of VR has been seen in the past few years with a consistently growing popularity. According to [22], 20.8 million people in the US used VR headset in 2019. This number is forecast to grow to 28.1 million by 2021. Statistics also show that the worldwide shipment of VR devices has grown over 60% in the past two years [18]. By 2025, the value of the VR market is expected to reach USD 87.97 billion, from USD 11.52 billion in 2019 [38]. While VR is traditionally used for recreational purposes, it is now rapidly permeating a variety of mission-critical applications ranging from e-business [2, 14, 71], healthcare [16, 63, 74], social networking [28, 37, 75], manufacturing [11, 25, 49], military training [41, 68, 73], and education [1, 7, 31].

In these applications, VR devices store their users' personal information, such as emails, photos, videos, and browsing history, as well as their online login accounts and passwords. Recently, online shopping and in-app purchases have emerged as important e-commerce opportunities for VR. For example, eBay launched a VR department store, where users can shop around in a virtual environment and make transactions online [9]. VR is also deemed as the future of social media interactions. In March 2020, Facebook started beta testing for its new VR social network "Horizons" where users engage with news content, share information, and entertain themselves in the virtual world by logging into Horizons using their accounts and passwords [52]. In the above scenarios, as the process of inputting data to current VR systems tends to be tedious, users may store their account and credit card information for auto-login and in-app purchase [54]. As a result, such practices may result in the security breach and even financial loss if the device is accidentally left unattended to people with ill purposes, including close friends and roommates. Therefore, the employment of user authentication mechanisms is crucial for VR devices. Only the owner or authorized users are able to unlock the device, while outliers are prohibited from access.

Unfortunately, user authentication on VR devices is yet far from well investigated. Current solutions, including password, digital PIN, and drawing pattern, mostly follow conventional approaches for general personal devices. However, these schemes have been proved vulnerable to shoulder-surfing attacks [21, 30, 32], as how password/PIN/pattern entered in VR device leaves little leverage to obfuscate the secret entry process. If the adversary is aware of the virtual digit board layout, it can easily decode hand movements to infer PIN inputs. The inference is even easier for the pattern-based authentication since the attacker only needs to track the hand movement trajectory without exquisite knowledge of the virtual board input design. Moreover, because a user's view is completely blocked from the physical world by the headset, it renders the user challenging to be aware of the presence of shoulder-surfing attackers.

To resist shoulder-surfing attacks, a shuffled keyboard has been proposed [3, 60]; the system adopts a new randomly generated keyboard layout each time a user intends to enter the credential. While leaving the key inference almost impossible, it sacrifices the authentication usability. Extra effort is incurred to the user in searching for keys on a shuffled keyboard. Recently, some novel user authentication methods for VR devices have been introduced. A couple of them focus on the improvement of the explicit knowledge-based authentication schemes, such as 3D password [29, 79] and spatial targets [27, 39]. These methods provide more robust authentication by implementing more complicated secret codes. However, they do not improve usability, if not further worsening it. For example, in [79], users are required to remember and enter a complicated 3D drawing pattern for authentication, which results in longer authentication time and a higher error rate. Some existing efforts employ the implicit biometrics to defend against shoulder-surfing attacks [6, 42, 50, 58]. Nonetheless, using biometrics alone suffer from irrevocability, which renders replay attacks a severe threat if even a single user's biometric sample is acquired by an attacker [24]. There are also some prior works on two-factor authentication [4, 10, 44]. So far, the existing solutions either rely on highly advanced equipment, such as a customized sensory headset with a number of electrodes to capture the brain signals[44], which is not readily available on current VR devices, or introduce heavy cognitive load that has users to perform complex and tedious authentication tasks.

1.2 Proposed Methodology

In this paper we propose *BlinKey*, a practical two-factor authentication scheme for VR devices that are equipped with eye trackers. Users authenticate themselves by blinking eyes following certain rhythm only known by themselves. It is a new passcode-style authentication. Rather than numbers, letters, or characters, users choose different beats/rhythms when blinking. Basically, a blinkey[1] can be easily created by the user, for example, by extracting some beats from his/her favorite songs or jingles. The knowledge-based feature of a blinkey is characterized by the timing of its blinks, which can be recorded by the eye tracker together with the system clock. Additionally, a blinkey is also characterized by its biometric features. We observe that how human pupils adapt to light after blinks, more specifically, the variation of pupil size, is unique for each person. As a blinkey is composed of multiple blinks, we then treat the pupil size variation, captured by the eye tracker, between blinks as a biological marker. Incorporating the above knowledge-based and biometric features, *BlinKey* serves as two-factor authentication to determine whether a user is legitimate or not.

BlinKey can be an ideal solution for user authentication on VR devices. First, it can effectively resist shoulder-surfing attacks. Unlike conventional PIN/password/pattern authentication, which requires users to hold the controller to enter credentials, *BlinKey* is simply performed by user blinking eyes. As the visual sight is blocked by the headset, it is impossible for the adversary to observe the passcode entry process. Second, it is convenient to perform. *BlinKey* is a hand-free authentication without imposing effort-demanding tasks. Third, as it involves both explicit knowledge and implicit biometric features, it is robust against attacks, such as guessing attacks and shoulder-surfing attacks. Although *BlinKey* only works for VR devices that are equipped with eye trackers, they are not a small population. To our knowledge, many VR headsets, such as HTC Vive Pro Eye [35], FOVE 0 [26], Pico Neo 2 [59], and Varjo VR-1 [72], are all in this category. These devices can therefore provide eye blinks and pupil size variations to the authentication unit on the device. We would like to note that integrating eye-tracking technology is a trend of VR headsets [64, 70], as it significantly improves user experience. For example, it helps VR headsets to simulate depth of field and focus, providing a more realistic and natural visual experience. *BlinKey*, as another user authentication scheme, can be employed for accessing both stand-alone devices and online accounts. This is also the case for many other user authentication schemes. For example, fingerprint-based authentication is widely adopted not only by a broad set of personal devices but also by some online services, such as online banking [23, 53].

BlinKey is composed of two phases. In the enrollment phase, users are asked to create their own blinkeys and enter them multiple times for the training purpose. During the login phase, the user simply enters the previously enrolled blinkey to unlock the device. If it matches the training samples, the user is authorized; otherwise, the access request is denied. To investigate the performance of *BlinKey*, we recruit 52 volunteers and collect 1306 blinkey samples from them. Classification accuracy is studied concerning different parameter settings. Based on the result, we implement our scheme on a commercial VR device with the parameter values that optimize the authentication performance. Another 43 participants are recruited. Multiple in-field experiments are conducted to evaluate the system performance in terms of attack resistance, time consumption, login attempts, the impact of user motions, and memorability, which outperform state-of-the-art solutions.

[1] In this paper, we utilize the Italian font *BlinKey* to represent the authentication scheme, while the regular font blinkey as the password itself.

Table 1. Comparison among different user authentication approaches on VR devices. (*) The work [30] discusses both PIN and pattern lock for VR.

System	Key space	Hand-free	Extra sensors	Accuracy	Security	Login time	Memorability
PIN [30]*	Low	No	No	High	Low	Short	High
Pattern lock [30]	Low	No	No	High	Low	Short	High
Shuffled keyboard [3]	Low	No	No	High	Medium	Medium	High
LookUnlock [27]	Low	Yes	No	High	Medium	Medium	Low
3D Pattern [79]	Medium	No	No	High	Medium	Short	Low
RubikAuth [48]	Medium	No	No	High	Medium	Short	Low
Hand gesture[42]	Medium	No	No	Low	Medium	Short	-
Brain biometrics [44]	High	Yes	Yes	Medium	Medium	Medium	-
Head movement [50]	Medium	Yes	No	Low	Medium	Long	-
SkullConduct [66]	Medium	Yes	Yes	Low	Medium	Short	-
Eye Movements [67]	Medium	Yes	Yes	Medium	Medium	Short	-
3D Password [4]	High	No	No	Medium	High	Medium	High
RubikBiom [47]	High	No	No	Medium	High	Short	Low
BlinKey (this work)	High	Yes	No	Medium	High	Medium	Medium

2 RELATED WORK

2.1 Knowledge-based Authentication

In recent years, how to authenticate users in VR devices has been increasingly explored in both computing and security research communities. George et al. carried out user study for the direct transfer of well-established user authentication concepts, including PIN and pattern lock, into VR [30]. Due to their vulnerability to shoulder-surfing attacks, a shuffled keyboard is proposed [3, 60]. Users enter their credentials on a virtual keyboard with a randomly generated layout each time. Yu et al. then develop a 3D pattern lock that creates an additional entropy for user's secret credentials [79]. Funk et at. [27] developed a graphical authentication mechanism based on gaze-tracking, called LookUnlock. The passcode consists of a set of virtual objects that a user's gaze focuses on in the correct sequence. A similar idea is adopted by [29, 39]. These schemes produce rather limited key space. For a passcode constructed by selecting 4 objects in a sequence from a total of 9 objects, the key space is merely $P(9, 4) = 3,024$, even smaller than that of the 4-digit PIN[2]. Moreover, it is not an easy task to remember the correct sequence of 4 objects. For example, according to the result [29], their 7-day recall rate is 74.1%. As shown in Section 6.3.3, this value is 89.6% for *BlinKey*. Mathis et al. proposed RubikAuth [48], where users select digits from a virtual 3D cube manipulated with a handheld controller. Following a similar idea, RubikBiom [47] further takes into account user behavioral biometric features such as hand movement when entering credentials from the virtual 3D cube. With the introduction of an additional layer of protection, RubikBiom is more robust against guessing attacks and shoulder-surfing attacks. As noted by the authors, both schemes require two-handed interactions which are inconvenient for users with motor disabilities. *BlinKey* is free from such a restriction for allowing users to enter their authentication credentials with eye blinks. Alsulaiman and Saddik [4] propose a 3D password that combines textual passwords and the user's behavior biometrics for entering the password.

[2] We will discuss in Section 7 that *BlinKey* offers the key space orders of magnitude higher than conventional passcodes, such as digit-PIN and password, of the same length.

2.2 Biometric Authentication

Unlike knowledge-based authentication, which is based on “what you know”, biometric authentication leverages “who you are” by looking into the unique biometrics that people are naturally born with. It has gained preference in certain situations due to its robustness against guessing attacks and shoulder-surfing attacks.

Gesture biometrics: Prior works [42, 50, 58] extract user’s distinctive biometric features from their head/hand/body movements for user authentication. These schemes require users to turn the head, bend the body in different directions, or throw/catch particular virtual objects. The involved actions may be awkward to perform especially in public places.

Gaze biometrics: Gaze tracking has recently been explored for user authentication. Existing solutions either examine the position or the content that a user is looking at or eye movement. The former is based on the hypotheses that each user’s gaze behaves uniquely when watching the screen [13, 61, 62]. These schemes rely on a large number of data samples to extract sufficient features for accurate authentication. As a result, they typically take more than one minute to authenticate a user. The second class of gaze-based authentication leverages the uniqueness of eye movement to fingerprint each user [8, 20, 33, 34, 40]. Relevant features include eye movement velocity and saccade latency. As pointed out by [81], these solutions suffer from irrevocability, which is in fact a common pitfall for many pure biometric-based authentication schemes. To address this issue, [67, 81] introduce the idea of random stimuli. As a result, the biometric features observed in each authentication trial become dynamic, leaving reply attacks infeasible. Nonetheless, they only work with precise eye movement tracking. For example, [67] requires a sampling rate of up to 500 Hz and tracking error within 0.4°, which cannot be met by current add-on eye trackers for VR devices.

Other biometrics. Schneegass et al. [66] present SkullConduct, a biometric system that uses bone conduction of sound through the user’s skull for user identification. A microphone is used to capture the skull vibration. Recently, Lin et al. [44] utilized responsive brainwaves when a user is presented with visual stimuli for authentication. Sophisticated electrodes should be integrated into VR headsets to capture the human brainwave.

2.3 Rhythm-Based Authentication

Only a few rhythm-based authentication schemes have been proposed so far. Wobbrock [78] developed an authentication system for single-key devices called “TapSongs”, which enables user authentication on a single “binary” sensor (e.g., button) by matching the rhythm of tap down/up events to a jingle timing model created by the user. A group authentication scheme, Thumprint [17], was proposed by Das et al., using the rhythm of a secret knock to authenticate a group of users, while each user’s expression of the secret is discernible. Chen [15] built a two-factor rhythm-based authentication scheme for multi-touch mobile devices. Recently, Hutchins et al. [36] developed a rhythm-based authentication scheme for wearable devices equipped with a touching sensor. TapMeIn [51] is another authentication method for smartwatches. On top of the secret tapping rhythm, it jointly considers biometric features, such as pressure and finger size of tapping. All these features are captured by smartwatch’s touching screen/sensors that are missing from current VR devices. Thus, TapMeIn is inapplicable to our case. Observing that how human pupils adapt to light after blinks, more specifically, the variation of pupil size, is unique for each person, we then treat it as a biological marker. Together with the user’s blinking rhythm, they are both captured by the eye tracker and serve as secret credentials for user authentication.

Summary. Table 1 provides a comprehensive comparison between some representative user authentication schemes for VR and *BlinKey*. The existing schemes are categorized into three groups, knowledge-based authentication (light gray), biometric-based authentication (medium gray), and two-factor authentication (dark gray). The comparison is made from the aspects of security (including “key space” and “security”) and usability (including “hand-free”, “extra sensors”, “accuracy”, “login time”, and “memorability”).

The salient advantage of knowledge-based authentication is mainly on its usability, with the highest accuracy and the lowest login time among the three groups. As user's passcodes are mostly entered by hand controllers, no extra sensor is needed. Nonetheless, these schemes have been criticized for their security, for example, vulnerable to shoulder-surfing and/or statistic attacks. This issue is partially resolved by some biometric-based authentication schemes. First, biometric features can barely be eavesdropped. Second, user's unique biometrics introduce a much larger key space. On the other hand, due to the hardware restriction, the explorable biometrics from VR devices are still limited so far. Some approaches require users to perform body/head/hand movement that can be readily captured by VR devices; some others rely on extra sensors, e.g., EMG and ECG sensors, to extract biometric features. There also have been a couple of two-factor authentication schemes that combine regular knowledge-based passcodes and user biometrics. Most of them exhibit better security performances than the other two. However, due to the involvement of behavior biometric features, which are dynamic even from the same user, their accuracy is degraded a little bit than knowledge-based schemes. Apparently, *BlinKey* belongs to the third group. Compared with [4, 47], it is entered hand free and thus friendly for users with motor disabilities. Moreover, as discussed in Section 7, rhythmic patterns produce a significantly larger key space than conventional PIN/password/pattern lock; so is *BlinKey* compared with [4, 47].

3 THREAT MODEL

The adversary's goal is to impersonate the legitimate user and successfully authenticate itself to the VR device. This work pertains to the discussion of the following commonly seen attacks.

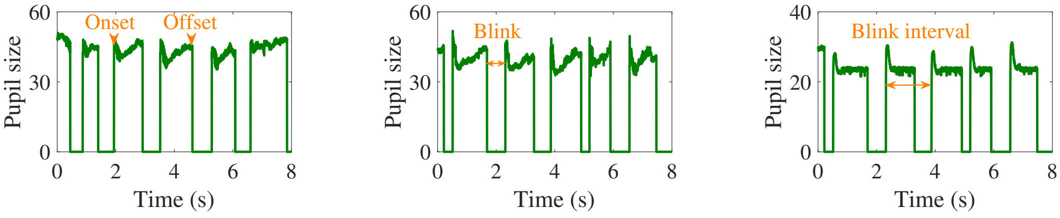
- *Zero-effort attack.* The adversary does not have any side information of the enrolled blinkey and tries to get authenticated by random guessing. It is also referred to as *guessing attack* in some other literature.
- *Statistical attack.* The adversary has access to a large volume of blinkeys and is aware of the set of features utilized by the scheme. It performs statistical analysis over the dataset and derives probability distribution over each feature. Then, the adversary forges synthetic blinkeys following the acquired distributions.
- *Shoulder-surfing attack.* The adversary is able to observe the authentication process while the victim is entering a blinkey. Then it mimics the legitimate user by repeating what it has observed.
- *Credential-aware attack.* This attack is even more powerful than the shoulder-surfing attack. We assume that the adversary has the full information of the legitimate user's secret blinking rhythm. The only difference from the shoulder-surfing attack is that the latter acquires blinking rhythm via visual observation.

We also make the following assumptions throughout the paper. The adversary cannot compromise the VR device or its connected server to access the user's blinkeys; otherwise, it renders secure user authentication design impossible. Due to the similar reason, the connection between the VR device and the server is also deemed secure.

4 BLINKEY CHARACTERIZATION AND FEATURES

4.1 Definition of BlinKey

A blinkey is composed of time instances stamped by the system clock when a user blinks in a self-designed rhythm, together with variations of pupil size exhibited between consecutive blinks. Both information can be recorded by the eye tracker. Figure 1 gives three exemplary blinkeys. Blink onset/offset indicates the moment that eyes are open/closed. An eye is deemed closed when its pupil size is measured zero and open otherwise. The length of a blinkey is simply the number of blinks it contains. For example, all the three blinkeys in Figure 1 are of length 6. We observe that the pupil size is not fixed between blinks. It experiences some fluctuations in the following procedures. When the eye is open, the eye tracker quickly captures the pupil's instantaneous size, which is at a large value. Then the pupil quickly adapts to ambient light by adjusting its diameter. After a short period, around dozens to hundreds of milliseconds, the pupil returns to a relatively stable status with micro-fluctuations. More importantly, we find that such a pupil's adaption pattern varies across people. Figure 1(b) and 1(c) demonstrate the same blinking rhythm performed by two users. While the rhythm is almost identical, the way how pupil size changes is clearly distinct between two trials. This is due to the pupil dilation/constriction that is controlled by the iris muscles with a biologically unique pattern [65]. Moreover, we also notice in Figure 1 that the pupil size variation pattern is consistent from the same user. Based on the above observation, we thus treat the pupil size variation between blinks as an additional dimension of features that fingerprint individuals.



(a) Blinkey 1

(b) Blinkey 2

(c) Blinkey 3

Figure 1. Three exemplary blinkeys.

4.2 Feature Selection

Since a blinkey consists of both knowledge-based features ("something you know") and biometric features ("something you are"), we are interested in identifying suitable feature set for user authentication.

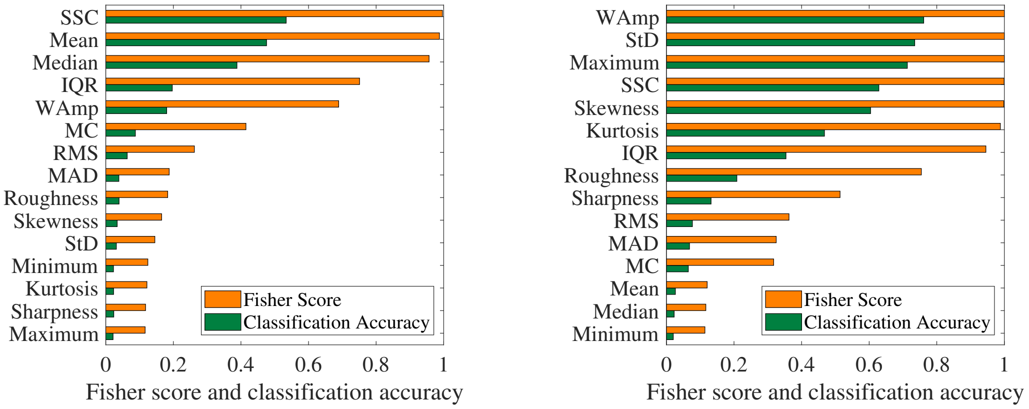
4.2.1 Knowledge-based Features. The knowledge-based features are the blink rhythm designed by the legitimate user. We mainly focus on the following three features *blink time instance*, *blink interval*, and *relative interval*.

- **Blink time instance.** The blink rhythm can be uniquely identified by a set of blink onsets and blink offsets, indexed by their timestamps, which are represented by two vectors $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \cdots, \alpha_n\}$ and $\boldsymbol{\beta} = \{\beta_1, \beta_2, \cdots, \beta_n\}$. Here, α_i and β_i are the timestamps for the i^{th} blink onset and offset, respectively, and n is the blinkey length, i.e., the number of blinks contained. For analysis consistency, we index the first blink onset as 0, $\alpha_1 = 0$. In other words, we deem the starting point of a blinkey as the moment when a user opens her eyes for the first time to perform her blinkey.
- **Blink intervals.** To characterize a blinkey's rhythm, we further extract the inter-onset intervals of a blinkey, defined as the time duration between two adjacent blink onsets, as shown in Figure 1: $\boldsymbol{\gamma} = \{\gamma_1, \gamma_2, \gamma_3, ..., \gamma_{n-1}\}$, where $\gamma_i = \alpha_{i+1} - \alpha_i$.
- **Relative intervals.** In actual scenarios, users' input speed may be influenced by their moods or other factors. Thus the time instance for each blink and their intervals may be different even for the same user entering a same blinkey. To take this into account, we introduce another feature, relative interval, which is defined as the ratio of a blink interval to its previous one: $\boldsymbol{\eta} = \{\eta_1, \eta_2, \eta_3, ..., \eta_{n-2}\}$, where $\eta_i = \frac{\gamma_{i+1}}{\gamma_i}$.

4.2.2 Biometric Features. As discussed above, the pupil size variation of each user can be treated as her biometric identifier. We now investigate the proper set of features to extract for authentication.

- **Fourier coefficients.** From the perspective of frequency analysis, the pupil size variation consists of components under different frequencies. To extract this information, we then apply the fast Fourier transform (FFT) over time-domain samples. The Fourier coefficient associated with each frequency component then serves as part of biometric features, $\boldsymbol{\phi} = \{\phi_1, \phi_2, \cdots, \phi_m\}$, where ϕ_i ($i \in [1, m]$) is the mean Fourier coefficient of the i^{th} frequency component. The larger coefficient of a higher frequency component a user produces, the more agile her pupils adapt to luminance. Computation and parameter setting details regarding Fourier coefficient extraction will be discussed in Section 5.3.

- **Statistical features.** In addition to Fourier coefficients, we further explore a few statistical features in both time and frequency domains that have been widely adopted in characterizing signals [5, 43, 46]. A set of candidate statistical features include, `Maximum`, `Minimum`, `Mean`, `Median`, `Root Mean Square (RMS)`, `Standard Deviation (StD)`, `Mean Absolute Deviation (MAD)`, `Kurtosis`, `Skewness`, `Interquartile Range (IQR)`, `Roughness`, `Sharpness`, `Mean Crossing (MC)`, `Willison Amplitude (WAmp)`, `Slope Sign Change (SSC)`, in time (T) and frequency (F) domains.



(a) Statistical features in time domain

(b) Statistical features in frequency domain

Figure 2. Fisher score and classification accuracy for statistical features.

Since not all of them play essential roles in our task, it is necessary to filter out non-critical ones. For this purpose, we calculate the *Fisher score* for each above feature. As one of the most commonly used supervised feature selection methods [45, 69], the Fisher score takes the inter-class variance and the in-class variance over the values of a given feature and computes their ratio. A higher ratio indicates that the distances between classes are much larger than those within the same class. *Classification accuracy*, an accuracy indicator of feeding each feature alone into the classifier, shows how well these features work for the classifier individually. Hence, we compute both the Fisher score and the classification accuracy for each statistical feature. Their results are shown in Figure 2(a) and Figure 2(b), respectively, in a descending order of a combination of both metrics. To facilitate the discussion, the Fisher score is normalized. We thus pick the top-ten best features, i.e., with the highest combined values, to constitute the statistical feature set $\boldsymbol{s}$ = {WAmp$_F$, StD$_F$, Maximum$_F$, SSC$_F$, Skewness$_F$, SSC$_T$, Mean$_T$, Kurtosis$_F$, Median$_T$, IQR$_F$}. The result is shown in Table 6.

The entire feature set to characterize a blinkey is then written as $\boldsymbol{f} = \{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\eta}, \boldsymbol{\phi}, \boldsymbol{s}\}$.

5 SYSTEM DESIGN

5.1 System Overview

Figure 3 shows an overview of the *BlinKey* system. It involves registration (or called enrollment) phase and login (or called testing) phase. For either phase, the workflow of data processing is summarized as follows. Authentication is turned on when a user awakens the screen, opens an app, or triggers a purchase interface.

Table 2. The selected statistical features.

Feature	Definition	Fisher Score	Accuracy
WAmp_F	The count of significant changes in the signal in frequency domain	0.9999	0.7614
StD_F	The extent of deviation for the signal amplitude in frequency domain	0.9998	0.7348
Maximum_F	The maximum value of the signal amplitudes in frequency domain	0.9999	0.7129
SSC_F	The count of slope sign changes in the signal in frequency domain	0.9992	0.6287
Skewness_F	The distortion of the signal values in frequency domain	0.9984	0.6042
SSC_T	The count of slope sign changes in the signal in time domain	0.9962	0.5339
Mean_T	The average value of the signal amplitudes in time domain	0.9873	0.4758
Kurtosis_F	The sharpness of the peak the signal curve in frequency domain	0.9882	0.4677
Median_T	The value that divides the signal amplitudes in half in time domain	0.9567	0.3882
IQR_F	The difference between the third and the first quartiles in frequency domain	0.9454	0.3538

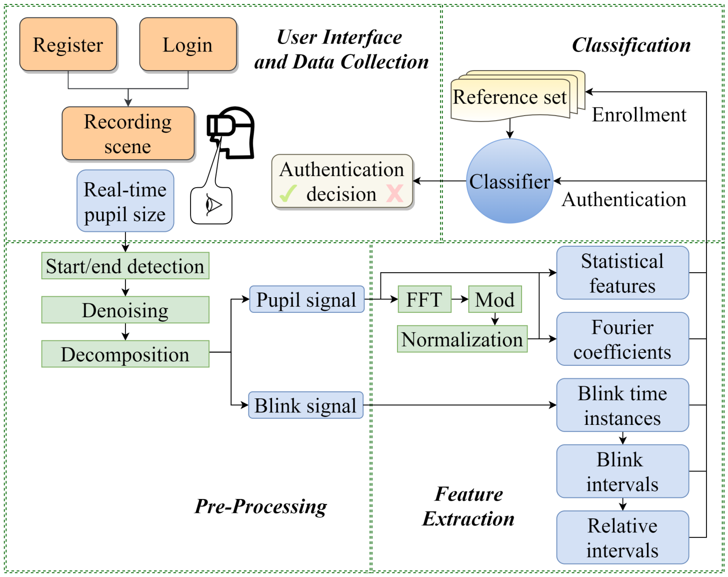


Figure 3. System overview.

In a pop-up virtual scene, the user is asked to blink in a self-designed pattern as an input blinkey. Once the authentication procedure is activated, the eye tracker keeps recording the user's real-time pupil size signals and transmit them to the server. The signal first passes the start/end detection module so as to segment the entire blinkey. The raw signal is then denoised and decomposed. Its outputs, including *blinking rhythm* and pupil size variations, are then fed into the feature extraction module to distill knowledge-based and biometric features. Finally, the classifier decides whether the given blinkey is legitimate or not.

5.2 Start and End Detection

A challenge of our approach lies in how to detect a blinkey, more specifically, identify its start and end points. This task is easy for authentication on regular personal devices, such as smartphones and tablets. For the case of pattern lock, the moment that a finger touches/leaves the screen is simply the start/end point of one trial. These moments can be accurately recognized by touching sensors embedded in the screen. For the case of password-based authentication, the end of one entry is explicitly indicated by tapping the enter/return key. Unfortunately, such hardware is unavailable at VR devices. One viable solution is to create a virtual enter/return key. However, it may incur extra effort for a user to interact with the virtual screen via a controller. Alternatively, we propose to have a user to indicate the start/end of a blinkey for closing the eyes a while, as shown in Figure 4. In this way, the moment that the user opens eyes for the first time after the long blink is treated as the start of the blinkey. Similarly, the moment that the user closes eyes right before the long blink is treated as the end of the blinkey.

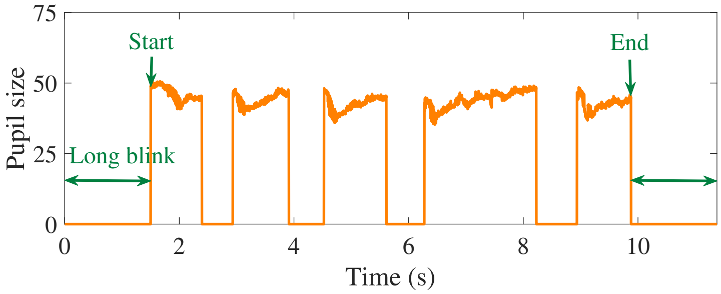


Figure 4. Illustration of start/end of a blinkey and the long blink.

The remaining question is to decide the duration for the long blink. Recall that a single blink is determined by the forceful closing of the eyelid. The system should be capable of differentiating between a long blink and a blink belonging to a blinkey or a spontaneous blink. We start by analyzing the statistics of spontaneous blinks based on the 434 blink samples collected from 22 volunteers. Its statistical distribution is plotted in Figure 5. We find that the duration of spontaneous blinks ranges from 0.09 to 0.26 second with its mean as 0.12 second, and the 95th percentile as 0.18 second. Our discovery concurs with the result of UCL Researcher [12], stating that the duration of a spontaneous blink is on average 0.1 - 0.15 second, as well as the result of Harvard Database of Useful Biological Numbers [55], stating that the duration of spontaneous blinks mainly ranges from 0.1 to 0.4 second.

We further investigate the statistics of voluntary blinks of blinkeys. Its distribution is derived based on another phase of data collection, where we acquired 1306 blinkey samples from 52 volunteers. The details of this data collection phase are provided in 5.5. We observe in Figure 5 that the 95th percentile exists at 1.95 seconds. Based on the statistical analysis, we set the duration of the long blink as 2.5 seconds. A longer duration will sacrifice the usability of authentication, while a shorter value renders the detection error-prone. As a note, a user does not have to estimate the exact 2.5 seconds before performing a blinkey, as long as the waiting duration is no less than the threshold. This requirement is easy to meet.

5.3 Pre-processing

The objective of this component is twofold, to filter out noise in the raw signal and to decompose the signal into ingredients that contain knowledge-based and biometric features separately.

5.3.1 Denoising. As shown in Figure 6, the raw signal is mainly composed of three components: voluntary blinks, spontaneous blinks, and the pupil adaptive variations between blinks. The useful information includes voluntary blinks and pupil adaptions. Spontaneous blinks are conducted in the pre-motor brain stem and happen without

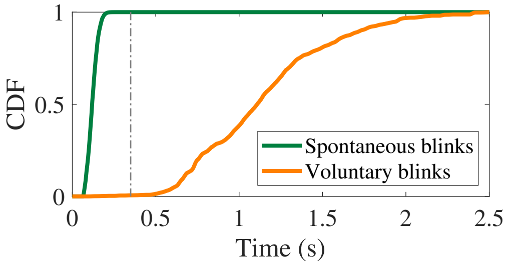


Figure 5. Statistical distribution of duration for spontaneous blinks and voluntary blinks.

conscious efforts, like breathing and digestion. It helps to spread the tear to all parts of the eyes and helps to keep them moist [76]. They are done involuntary and distinct from the voluntary blinks in a blinkey. As the involvement of spontaneous blinks brings the noise to the feature extraction and thus authentication accuracy, the goal of this phase is to eliminate spontaneous blinks from the raw signal.

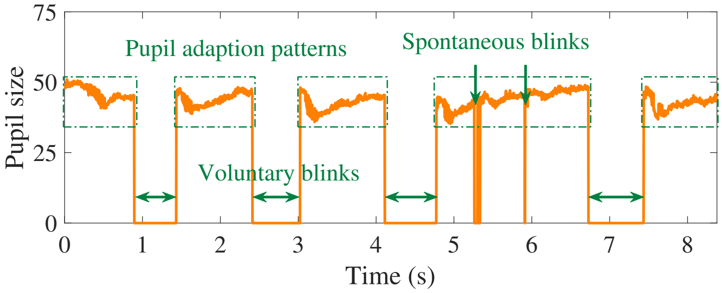


Figure 6. The raw signal of a blinkey mainly consisting of voluntary blinks, pupil adaptions, and spontaneous blinks.

As shown in Figure 5, the statistical analysis indicates that the time duration of spontaneous blinks and voluntary blinks is clearly distinct from each other. It is noticed that the former mostly falls within the range from 0.09 to 0.26 second, while the latter is between 0.45 and 2.5 seconds. Motivated by this observation, we thus set a detection threshold at 0.35 second. For a blink whose duration is beyond this value, it is treated as a voluntary one; otherwise, it is a spontaneous one, which is eliminated from the raw signal. Meanwhile, it is infeasible to directly set their associated pupil size to 0's, as it will pollute the blinkey's features. Instead, we apply the *spline interpolation* [77]. As a common interpolation technique, it estimates missing data using a mathematical function that minimizes overall surface curvature. In our case, pupil sizes of spontaneous blinks are treated as the missing data and interpolated accordingly. In this way, we eliminate spontaneous blinks from the signal while preserving the blinkey features.

5.3.2 Decomposition. The goal of decomposition is to extract from the denoised signal user's *blinking rhythm* and *segments*, which carry knowledge-based features and biometric features of a blinkey, respectively. The decomposition facilitates the feature extraction next. As shown in Figure 7, a *segment* is simply the set of non-zero pupil size values between two consecutive blinks. A segment reflects the user's pupil variations after each voluntary blink. The decomposition is done by detecting all onsets and offsets in a blinkey. Since humans perform eyelid opening and closure rapidly, it leads to sharp rises and drops in the observed pupil size. Therefore, the detection of onsets and offsets can be accomplished via simple edge detection algorithms.

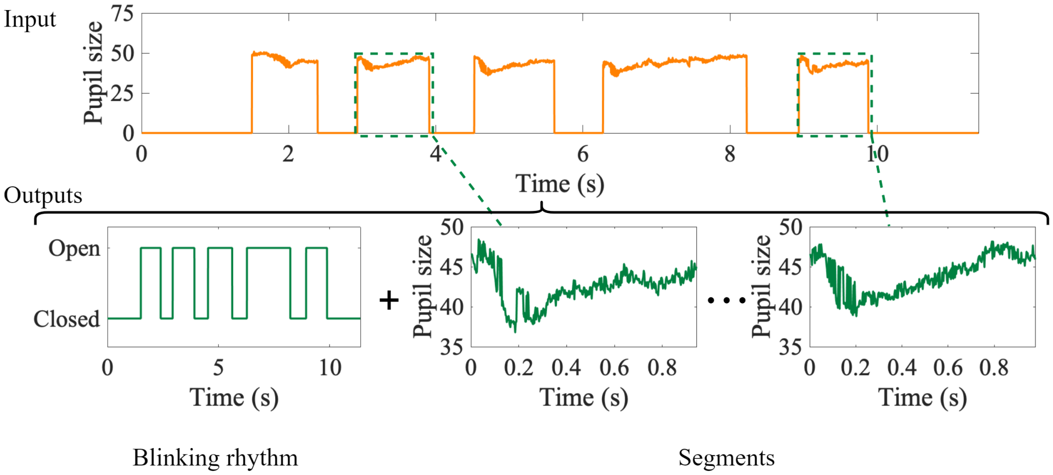


Figure 7. Illustration of decomposition. The input (top) is denoised signals and the output (bottom) is *blinking rhythm* and *segments*.

5.4 Feature Extraction

Once *blinking rhythm* and *segments* are ready, we are ready to extract from them desired features.

Knowledge-based features can be directly derived from the *blinking rhythm*. Specifically, we first obtain the time instances of onsets and offsets, i.e., $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. Then, the blink interval set $\boldsymbol{\gamma}$ and relative interval set $\boldsymbol{\eta}$ are calculated following their definitions in Section 4.2.1.

Biometric features involved in our scheme are classified into time-domain features and frequency-domain features. For the former, they include SSC, Mean, Median, etc. They can be computed based on time-series samples from one blinkey entry following definitions of these metrics. For the latter, they include WAmp, StD, Maximum, etc. As the first step, we employ FFT to decompose time-domain samples into their constituent frequencies. The frequency-domain representation can decompose complicated pupil size variations into periodic components that time-domain analysis cannot realize. FFT is applied over each segment. Before that, we first employ zero-padding to ensure that each segment has the same length of 1024 data points. The reason for choosing 1024 is twofold. First, FFT works most efficiently for a signal with length a power of 2 since it recursively folds the size at each step. Second, we observe from the 1306 collected samples that all segments last within 5 seconds. Given the sampling rate as 200 Hz in our system, every segment is sampled into 1000 data points the maximum. Based on the above discussion, we pad the segment into 1024 data points. Once the Fourier coefficients are derived for each segment, we take their average over all segments for each frequency component. It then produces the Fourier coefficient feature $\boldsymbol{\phi}$. Frequency-domain statistical features are computed following a similar method for time-domain statistical features.

5.5 Classification

Once features of a blinkey are extracted following previous steps, the remaining task is to apply classification methods for user authentication, i.e., to discriminate the legitimate user and imposters. Two common classification methods are considered, one-class Support Vector Machine (SVM) and K-Nearest Neighbors (k-NN). To determine which one best serves our system, we conduct comprehensive evaluations based on our dataset consisting of

Table 3. Demographics of volunteers in the phase-I study.

Gender	No.	Age range	No.	Eye color	No.	Eye wear type	No.
Female	36	18-23	19	Black	29	None	25
Male	16	24-29	28	Brown	14	Colorless glasses	21
		30-35	5	Hazel	8	Colorless contact lenses	4
				Green	1	Colored contact lenses	2

1306 blinkeys from 52 volunteers. These volunteers are all college students, including 36 females and 16 males. The classification performances are examined through the following metrics.

- False Rejection Rate (FRR). The probability that a legitimate user is rejected by the system. It is calculated as the ratio of the number of a legitimate user's incorrect authentications to the total number of attempts.
- False Acceptance Rate (FAR). The probability that an impostor is given access, computed as the ratio of the number of an impostor's authentication attempts that are accepted by the system to the total number of attempts.
- Equal Error Rate (EER). The point at which FRR and FAR are equal, by adjusting parameter values.

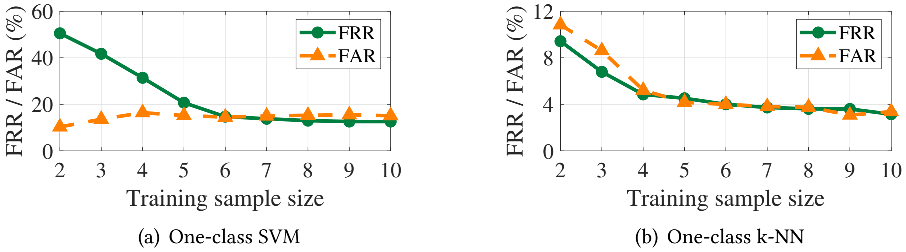
Note that FRR reflects the user convenience in our system; a lower FRR implies that a legitimate user can successfully unlock the VR device at a higher probability. FAR reflects the security aspect; a lower FAR implies that the imposter will be denied at a higher probability. It is worth noting that two blinkeys are deemed different with different lengths. For example, if the legitimate blinkey has a length of 6, then any testing input with a different length will be rejected immediately. Hence, in the following we only focus on the classification over blinkeys of the same length.

To investigate the performance of *BlinKey*, we performed two user studies. In phase I, the objective is to collect blinkeys created by different users so as to carry out statistical analysis and classification model selection as discussed here. In phase II, a prototype of *BlinKey* is built. We then conduct a series of in-field experiments to evaluate the security and utility of our system which will be covered in the next section.

In the phase-I user study, a specialized app is developed and implemented on the VR system to facilitate the data collection. A total of 52 volunteers are recruited. They are all college students aged from 18 to 35. Among them, there are 36 females and 16 males. Their demographic details are provided in Table 3. Before the data collection, they are explained how *BlinKey* works. Each volunteer is asked to design several different blinking patterns. For each pattern, a video of the volunteer's pupils is recorded by an eye tracker in the VR headset. Afterwards, they are shown the detected pupil size signal and are asked to manually mark the voluntary blinks from the spontaneous blinks according to their self-designed patterns. In total, we obtain 1306 samples containing 7,528 voluntary blinks and 3,673 spontaneous blinks. The collected dataset is used to derive statistics of blinkeys. Besides, we also aim to identify suitable parameters for the classifier.

5.5.1 Support Vector Machine. One-class SVM has been successfully applied to a number of classification problems. It generalizes the idea of finding an optimal hyper-plane in high-dimensional space to perform classification. Compared to other classification methods, it has advantages in implementation simplicity and efficiency in dealing with high-dimensional, non-linear datasets. Here, one-class SVM is implemented with the Radial Basis Function (RBF) kernel.

The number of training samples is an important indicator of classification performances. We tune the value from 2 to 10 and evaluate its impact. Figure 8(a) shows the authentication accuracy with respect to the training sample size. We observe that the FRR is as high as 50.5% with only 2 training samples. It drops quickly to 14.6% under 6 training samples. It mildly decreases to 12.6% as the training sample size grows to 10. The FAR grows



(a) One-class SVM

(b) One-class k-NN

Figure 8. FRR and FAR under different training sample sizes. (a) One-class SVM. (b) One-class k-NN.

from 10.3% with 2 training samples and keeps relatively stable around 15.0% as the training sample size increases to 10. The minimum EER 14.6% is achieved with 6 training samples.

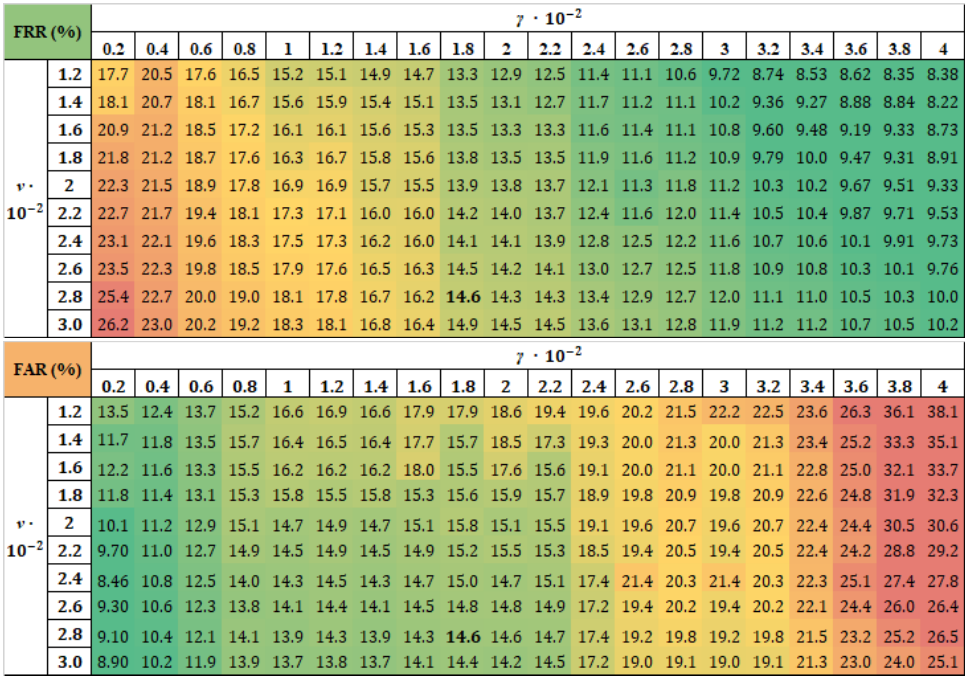
We further evaluate the performance of SVM with respect to the kernel coefficients γ and ν in Figure 9. Here, γ is the standard deviation of the kernel function. It influences the decision boundary qualitatively. As γ grows, FAR increases while FRR decreases, which means both legitimate users and impostors are more likely to get authenticated. In fact, for a larger γ, the decision criteria tend to be relaxed to avoid the hazard of overfitting. For a smaller γ, the decision boundary tends to be strict and sharp. In contrast to the former situation, it tends to overfit. The parameter ν is an upper bound on the fraction of margin errors and a lower bound of the fraction of support vectors relative to the total number of training samples. For example, ν=0.01 means that at most 1% of the training samples are misclassified (at the cost of a small margin, though) and at least 1% of the training samples are support vectors. Hence, as shown in Figure 9, a larger ν leads to a lower FAR but at the cost of a higher FRR. Combining the results above, EER reaches its lowest point at 14.6% when training sample size, γ, and ν are set to 6, 0.018, and 0.028, respectively. Hence, one-class SVM produces unsatisfactory authentication accuracy in our system.

5.5.2 K-Nearest Neighbors. Another classification method under consideration is k-NN. It measures the similarity between the testing sample and training samples. The similarity is represented by the *Manhattan distance*. If the score is below the threshold, the testing sample is considered a legitimate input; otherwise, it is an outlier.

We first examine the classification accuracy with respect to the training sample size. As shown in Figure 8(b), both FRR and FAR decreases with a larger training sample size. The detection accuracy improvement becomes insignificant, with 6 or more training samples. To balance between accuracy and usability, we use 6 samples to train the model. Comparing between Figure 8(a) and Figure 8(b), we find k-NN produces a much lower error rate. Given 6 training samples, EER of SVM and k-NN is 14.6% and 4.0%, respectively. The latter is less than 1/3 of the former.

We then investigate the impact of two critical parameters, k, the number of neighbors to select, and α, the threshold from the Manhattan distance matrix. A larger k indicates that more neighbors are taken into the calculation of the classification score. A larger α means a testing sample is more likely to be accepted legitimate. The results demonstrated in Figure 10 meet our expectations. A larger α, i.e., a loose detection rule, results in lower FRR but a higher FAR. As we increase the value of k, the classification becomes more stable due to majority voting/averaging, and thus, is more likely to make more accurate detection. Nonetheless, as k is beyond a certain value, we will witness an increasing number of errors as the value of k is pushed too far. As shown in Figure 10, the lowest EER exists at 4.0% with k = 3 and α = 1.0.

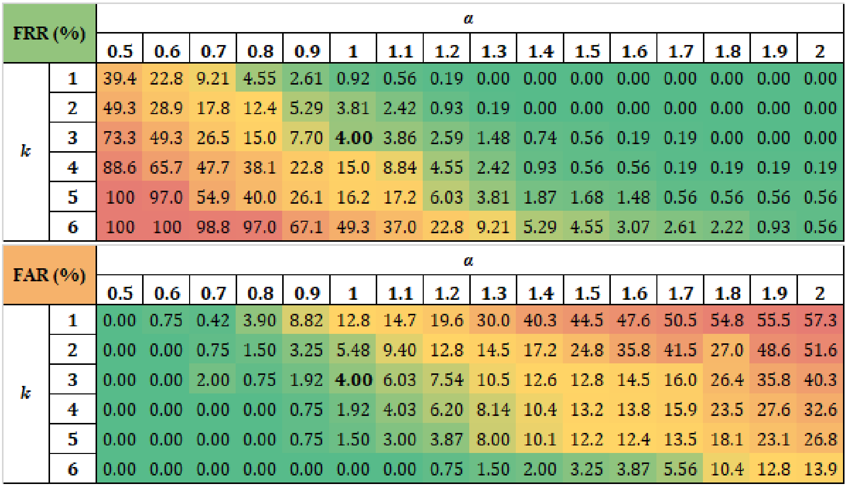
5.5.3 Other Classifiers. We further examine the classification accuracy of convolutional neural networks (CNN) and random forests (RF) in the latest version. Specifically, one-class CNN and one-class RF are considered. The former is based on CNN for one-class classification problems. Its idea is to use a zero centered Gaussian noise in



FRR (%)		$\gamma \cdot 10^{-2}$																			
		0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2	2.2	2.4	2.6	2.8	3	3.2	3.4	3.6	3.8	4
$\nu \cdot 10^{-2}$	1.2	17.7	20.5	17.6	16.5	15.2	15.1	14.9	14.7	13.3	12.9	12.5	11.4	11.1	10.6	9.72	8.74	8.53	8.62	8.35	8.38
	1.4	18.1	20.7	18.1	16.7	15.6	15.9	15.4	15.1	13.5	13.1	12.7	11.7	11.2	11.1	10.2	9.36	9.27	8.88	8.84	8.22
	1.6	20.9	21.2	18.5	17.2	16.1	16.1	15.6	15.3	13.5	13.3	13.3	11.6	11.4	11.1	10.8	9.60	9.48	9.19	9.33	8.73
	1.8	21.8	21.2	18.7	17.6	16.3	16.7	15.8	15.6	13.8	13.5	13.5	11.9	11.6	11.2	10.9	9.79	10.0	9.47	9.31	8.91
	2	22.3	21.5	18.9	17.8	16.9	16.9	15.7	15.5	13.9	13.8	13.7	12.1	11.3	11.8	11.2	10.3	10.2	9.67	9.51	9.33
	2.2	22.7	21.7	19.4	18.1	17.3	17.1	16.0	16.0	14.2	14.0	13.7	12.4	11.6	12.0	11.4	10.5	10.4	9.87	9.71	9.53
	2.4	23.1	22.1	19.6	18.3	17.5	17.3	16.2	16.0	14.1	14.1	13.9	12.8	12.5	12.2	11.6	10.7	10.6	10.1	9.91	9.73
	2.6	23.5	22.3	19.8	18.5	17.9	17.6	16.5	16.3	14.5	14.2	14.1	13.0	12.7	12.5	11.8	10.9	10.8	10.3	10.1	9.76
	2.8	25.4	22.7	20.0	19.0	18.1	17.8	16.7	16.2	**14.6**	14.3	14.3	13.4	12.9	12.7	12.0	11.1	11.0	10.5	10.3	10.0
	3.0	26.2	23.0	20.2	19.2	18.3	18.1	16.8	16.4	14.9	14.5	14.5	13.6	13.1	12.8	11.9	11.2	11.2	10.7	10.5	10.2

FAR (%)		$\gamma \cdot 10^{-2}$																			
		0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2	2.2	2.4	2.6	2.8	3	3.2	3.4	3.6	3.8	4
$\nu \cdot 10^{-2}$	1.2	13.5	12.4	13.7	15.2	16.6	16.9	16.6	17.9	17.9	18.6	19.4	19.6	20.2	21.5	22.2	22.5	23.6	26.3	36.1	38.1
	1.4	11.7	11.8	13.5	15.7	16.4	16.5	16.4	17.7	15.7	18.5	17.3	19.3	20.0	21.3	20.0	21.3	23.4	25.2	33.3	35.1
	1.6	12.2	11.6	13.3	15.5	16.2	16.2	16.2	18.0	15.5	17.6	15.6	19.1	20.0	21.1	20.0	21.1	22.8	25.0	32.1	33.7
	1.8	11.8	11.4	13.1	15.3	15.8	15.5	15.8	15.3	15.6	15.9	15.7	18.9	19.8	20.9	19.8	20.9	22.6	24.8	31.9	32.3
	2	10.1	11.2	12.9	15.1	14.7	14.9	14.7	15.1	15.8	15.1	15.5	19.1	19.6	20.7	19.6	20.7	22.4	24.4	30.5	30.6
	2.2	9.70	11.0	12.7	14.9	14.5	14.9	14.5	14.9	15.2	15.5	15.3	18.5	19.4	20.5	19.4	20.5	22.4	24.2	28.8	29.2
	2.4	8.46	10.8	12.5	14.0	14.3	14.5	14.3	14.7	15.0	14.7	15.1	17.4	21.4	20.3	21.4	20.3	22.3	25.1	27.4	27.8
	2.6	9.30	10.6	12.3	13.8	14.1	14.4	14.1	14.5	14.8	14.8	14.9	17.2	19.4	20.2	19.4	20.2	22.1	24.4	26.0	26.4
	2.8	9.10	10.4	12.1	14.1	13.9	14.3	13.9	14.3	**14.6**	14.6	14.7	17.4	19.2	19.8	19.2	19.8	21.5	23.2	25.2	26.5
	3.0	8.90	10.2	11.9	13.9	13.7	13.8	13.7	14.1	14.4	14.2	14.5	17.2	19.0	19.1	19.0	19.1	21.3	23.0	24.0	25.1

Figure 9. FRR and FAR with respect to γ and ν under one-class SVM.



FRR (%)		α															
		0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2
k	1	39.4	22.8	9.21	4.55	2.61	0.92	0.56	0.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	2	49.3	28.9	17.8	12.4	5.29	3.81	2.42	0.93	0.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	3	73.3	49.3	26.5	15.0	7.70	**4.00**	3.86	2.59	1.48	0.74	0.56	0.19	0.19	0.00	0.00	0.00
	4	88.6	65.7	47.7	38.1	22.8	15.0	8.84	4.55	2.42	0.93	0.56	0.56	0.19	0.19	0.19	0.19
	5	100	97.0	54.9	40.0	26.1	16.2	17.2	6.03	3.81	1.87	1.68	1.48	0.56	0.56	0.56	0.56
	6	100	100	98.8	97.0	67.1	49.3	37.0	22.8	9.21	5.29	4.55	3.07	2.61	2.22	0.93	0.56

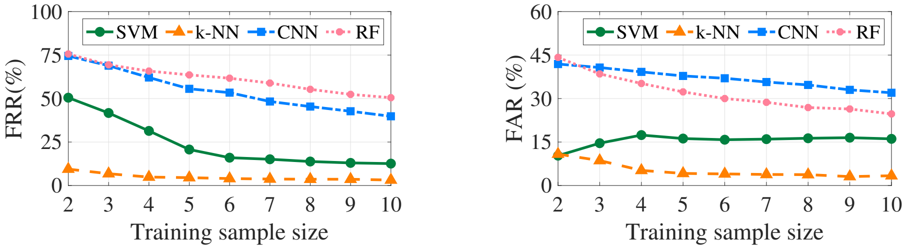
FAR (%)		α															
		0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2
k	1	0.00	0.75	0.42	3.90	8.82	12.8	14.7	19.6	30.0	40.3	44.5	47.6	50.5	54.8	55.5	57.3
	2	0.00	0.00	0.75	1.50	3.25	5.48	9.40	12.8	14.5	17.2	24.8	35.8	41.5	27.0	48.6	51.6
	3	0.00	0.00	2.00	0.75	1.92	**4.00**	6.03	7.54	10.5	12.6	12.8	14.5	16.0	26.4	35.8	40.3
	4	0.00	0.00	0.00	0.00	0.75	1.92	4.03	6.20	8.14	10.4	13.2	13.8	15.9	23.5	27.6	32.6
	5	0.00	0.00	0.00	0.00	0.75	1.50	3.00	3.87	8.00	10.1	12.2	12.4	13.5	18.1	23.1	26.8
	6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.75	1.50	2.00	3.25	3.87	5.56	10.4	12.8	13.9

Figure 10. FRR and FAR with respect to α and k under one-class k-NN.

the latent space as the pseudo-negative class and train the convolutional network using the cross-entropy loss to learn a good representation and the decision boundary for a given class [57]. CNN has been widely applied to computationally complex classification tasks, such as image defect detection [80] and face verification [56].

One-class RF is a method based on a random forest algorithm and an original outlier generation procedure that makes use of classifier ensemble randomization principles [19]. The basic idea is to to use some randomization principles of ensemble learning methods to sub-sample the number of features and the number of training target instances to make possible the generation of outliers from the computation perspective, and to make use of the information given by the target samples to adapt accordingly the outlier distribution. Compared to CNN, it is faster to perform and requires fewer data samples.

As shown in Figure 11, given the same training sample size, k-NN achieves the lowest FRR and FAR among the four classifiers, while CNN and RF exhibit the worst performance. This is because the latter two generally require a large dataset to properly train their models. An empirical implication indicates that it typically takes at least 5,000 samples to train CNN with 10 or more layers and hundreds of neurons for satisfying accuracy in applications like image classification. Similarly, the training sample size is around 500 to train RF for relatively good performance in a classification problem. On the other hand, only 6 samples are needed for k-NN to obtain EER as low as 4.0%. It indicates that k-NN attains a promising authentication accuracy with much fewer training samples, especially compared with CNN and RF. Besides, with simple structures, k-NN and SVM consume fewer computation resources for training and testing than the other two. Thus, they are deployable to a wide spectrum of VR devices with heterogeneous resource capacities.



(a) FRR

(b) FAR

Figure 11. Authentication accuracy comparison among four classifiers.

To sum up, k-NN outperforms SVM, CNN, and RF in terms of classification accuracy, given the same training sample size in our case. More importantly, our design acquires limited training samples, as few as 6. Hence, the enrollment of a blinkey can be performed efficiently. Besides, our approach also outperforms [50, 81], two recently proposed user authentication schemes for VR devices, in terms of authentication accuracy. For [50], its EER is 7.4%. For [81], its EER is 6.9%. Both are higher than ours.

6 PERFORMANCE EVALUATION

6.1 Prototype Implementation & Experiment Setup

As a proof-of-concept implementation, we develop the prototype of *BlinKey* on an HTC Vive Pro head-mount device, connected to a local server[3] running SteamVR to support the VR environment. We install a Pupil Labs eye tracker in the VR device to record the real-time pupil size. The sampling rate is set to 200 Hz, i.e., pupil size

[3] The local server is a typical arrangement for the tethered VR headset, which our prototype device HTC Vive Pro belongs to. The local server is not a required element for *BlinKey*. Although our prototype makes use of the local server to do classification, the computation load is pretty light. Instead of any resource-demanding classification models, such as neural networks, *BlinKey* employs the light-weight k-NN. Thus, the computation can be practically supported on standalone VR devices with on-board computing units.

Table 4. Distribution of volunteer information.

Gender	No.	Age range	No.	Eye color	No.	Eye wear type	No.	Experience	No.
Female	16	18-23	15	Black	20	None	23	None	25
Male	27	24-29	26	Brown	18	Colorless glasses	17	Limited	14
		30-35	2	Hazel	3	Colorless contact lenses	2	Proficient	4
				Blue	2	Colored contact lenses	1		

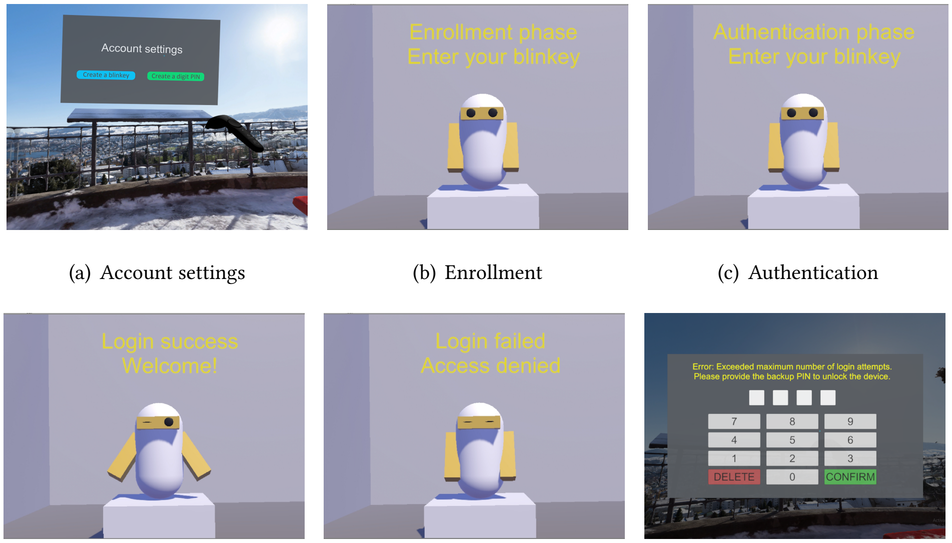
samples are collected every 5 milliseconds. The collected data are fed into the server through ZeroMQ application program interface (API). All the functions, such as start/end detection, pre-processing, feature extraction, and classification, are implemented in Unity, a cross-platform engine for VR games. As observed in Section 5.5, k-NN yields better accuracy than SVM in our system. Hence, we implement the former as the classifier in our prototype. The training sample size is set to 6, which means a user is asked to enter her blinkey 6 times in the enrollment phase. We set the parameters k as 3 and α as 1.0, since the k-NN demonstrates the best authentication accuracy with this setting. For comparison purposes, we also implement the basic PIN and pattern lock authentication schemes on the same VR device. Their corresponding passcodes are entered using controllers paired with the device.

To evaluate the security and usability of *BlinKey*, another 43 participants are recruited to conduct experiments. Among them, 13 volunteers also participated in the prior data collection session. The distribution of the participants' information is shown in Table 4. At the beginning of the experiment, the basic idea of *BlinKey* is explained to the participants. They are then trained on how to correctly enter a blinkey. Thereafter, they are asked to create their own blinkeys.

Screenshots of the user interface (UI) of our prototype are shown in Figure 12. UI is implemented in a virtual scene in Unity and displayed in the VR headset to guide users for enrollment and authentication. For the blinkey enrollment, we follow the basic steps of how an iPhone enrolls a user's fingerprints. Specifically, when legitimate users boot their new VR devices for the first time, they are guided to the process of account setting. As one of the steps, users are prompted to enroll their blinkeys (see Figure 12(a)). Users are asked to enter their blinkeys repeatedly until 6 valid samples have been collected (see Figure 12(b)). If a user tends to enroll another blinkey, the user is first required to provide the existing blinkey correctly. Then the rest steps similarly follow the ones for the initial account setup. The authentication is automatically triggered as a user puts on the VR headset, initiates an online purchase, or tries to log into her Internet account. A dialog box pops up, asking the user to enter her valid blinkey (as shown in Figure 12(c)). Based on the input, the classifier decides whether this entry is from the legitimate user: if yes, the access is granted (see Figure 12(d)); otherwise, the access is denied with an error message shown on the screen (see Figure 12(e)). If denied, a user can re-enter her blinkey until reaching the maximum number of attempts allowed, say 5. Then, the account is temporarily locked, and the recovery process is invoked (see Figure 12(f)).

6.2 Robustness Against Attacks

The adversary's goal is to impersonate a legitimate user and successfully get authenticated to the VR device. We assume that the adversary has physical access to the device. In practice, such physical access can be gained in ways such as a thief stealing a device, finders finding a lost device, and a roommate temporarily accessing a device when the owner is taking a shower. In the experiment, we consider the following types of attacks: *zero-effort attacks*, *statistical attacks*, *shoulder-surfing attacks*, and *credential-aware attacks*.



Account settings

Enrollment phase
Enter your blinkey

Authentication phase
Enter your blinkey

Login success
Welcome!

Login failed
Access denied

Error: Exceeded maximum number of login attempts.
Please provide the backup PIN to unlock the device.

(a) Account settings (b) Enrollment (c) Authentication

(d) Success (e) Failed (f) Locked out

Figure 12. Screenshots of UI for *BlinKey*.

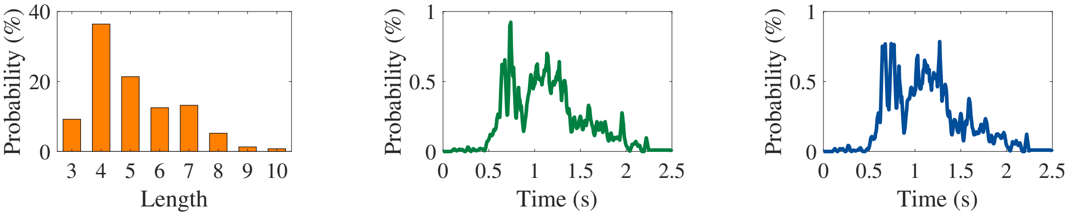
Table 5. Success rate of zero-effort attacks under different blinkey lengths.

Blinkey length	3	4	5	6	7	8	9	10
FAR (%)	8.1	4.4	3.4	1.9	0	0	0	0

6.2.1 Zero-effort Attacks. Zero-effort attacks may be the most common type of attacks against an authentication system, where the attacker guesses the secret or tries the authentication procedure without much knowledge of the legitimate password. In our case, each volunteer (attacker) is asked to randomly pick blinkeys without any prior knowledge of the legitimate one and tries to pass the authentication by chance. Up to five authentication attempts can be made. An attack is considered to succeed if any one of them passes the authentication.

Table 5 shows the success rate of zero-effort attacks, which is directly the FAR of our mechanism. Among 1306 collected blinkeys, all of them have the length between 3 and 10. Hence, we conduct tests over blinkey with their lengths falling within this range. Clearly, the blinkey length plays a critical role in the success rate of zero-effort attacks. The longer a blinkey is, the less possible it can be compromised by an adversary. Particularly, if the length is 7 or longer, the success rate drops to zero. Therefore, in the practical implementation of *BlinKey*, the system can impose a hard constraint over a valid blinkey's minimum length, say 7, to defeat zero-effort attacks.

6.2.2 Statistical Attacks. This type of attack assumes that the adversary has access to a abroad set of user's blinkeys. This type of attackers employ knowledge obtained from the statistics of a group of blinkeys as hints to generate authentication attempts. The basic approach is to estimate the feature distribution and then use the most probable feature values to generate the forgery. In the experiment, we use the 1306 collected blinkey samples and produce a set of forgery blinkeys as follows. We first randomly select a length following the probability distribution of all blinkey lengths, as illustrated in Figure 13(a). Then we randomly choose values for each eye



(a) Blinkey length (b) Blink duration (c) Opening-phase duration

Figure 13. Probability distributions of knowledge-based features.

blink and open following their probability distributions derived from our dataset. Figure 13(b) and Figure 13(c) depict these two distributions. Finally, a set of 150 forgery blinkeys is generated in this process.

Table 6. Success rate of statistical attacks under different blinkey lengths.

Blinkey length	3	4	5	6	7	8	9	10
FAR (%) of statistical attacks	5.2	6.4	2.8	2.4	1.8	0	0	0

An attacker is randomly assigned multiple forgery blinkeys and tries to get authenticated by repeating them. Hence, attackers use their own pupils and thus biometric features to launch the attack. Table 6 shows the success rate, i.e., FAR, of statistical attacks of *BlinKey*. The attacker's success rate drops to 0 for blinkeys when their lengths reach 8. Notably, statistic analysis does not grant the attacker much privilege over zero-effort attacks.

We further the variation pattern of *BlinKey*, specifically, the rhythm pattern distribution of blinkeys (without considering the biometric features) based on our dataset. Its purpose is to examine if users tend to choose similar blinking rhythms which would render the scheme vulnerable to statistical attacks. As shown in Table 7, we list the top-13 most frequently used blinkeys by analyzing 1306 valid enrollments in the dataset. 11 of them are the same, indexed as #1 blinkey, with their frequency calculated as 2.1%. Besides, there are also duplicates for #2–#10 blinkeys, with their occurrence frequencies as 1.5%, 0.9%, 0.8%, 0.6%, 0.6%, 0.4%, 0.4%, 0.4%, and 0.4%, respectively.

Table 7. Frequency of blinkeys from collected dataset.

Index	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13
Frequency	2.1%	1.5%	0.9%	0.8%	0.6%	0.6%	0.4%	0.4%	0.4%	0.4%	0.2%	0.2%	0.2%

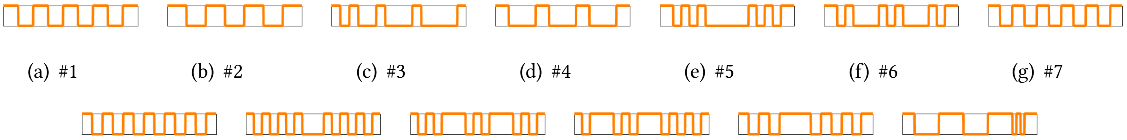
It implies that users are less likely to choose the same blinking pattern. Therefore, attackers can barely obtain useful information from the statistic analysis over a set of blinkeys. We acknowledge that our dataset is limited in its size, with only 1306 blinkeys. Still, our analysis partially reflects the blinking pattern distribution in practice. Compared with regular digit-PIN and password, a blinkey can be characterized by a more rich set of features, including tapping time instances, tapping intervals, relative intervals, and even pupil size variations. All these factors make *BlinKey* robust against statistical attacks.

We further visualize in Figure 14 the top 13 commonly adopted blinkey patterns that are presented in Table 7. As shown, the patterns that exhibit uniform rhythms (#1, #2, #4, #7, and #8) or symmetric rhythms (#5, #6, #9, and #10) are more likely to be adopted. Such a phenomenon is also observed in PINs; the commonly picked

Table 8. Success rate of credential-aware attacks on *BlinKey*, PIN, and pattern lock.

Length	3	4	5	6	7	8	9	10
BlinKey	16.6%	25.4%	19.7%	15.5%	14.2%	10.6%	7.9%	4.4%
PIN	100%	100%	97.1%	100%	100%	100%	100%	100%
Pattern lock	100%	100%	100%	100%	100%	99.3%	97.1%	96.8

PINs include 000000, 010101, etc., which share similar properties above. Note that Blinkey is a two-factor user authentication that also involves biometric features. Hence, it effectively avoids PIN and password pitfalls caused by popular credential selections.



(a) #1 (b) #2 (c) #3 (d) #4 (e) #5 (f) #6 (g) #7

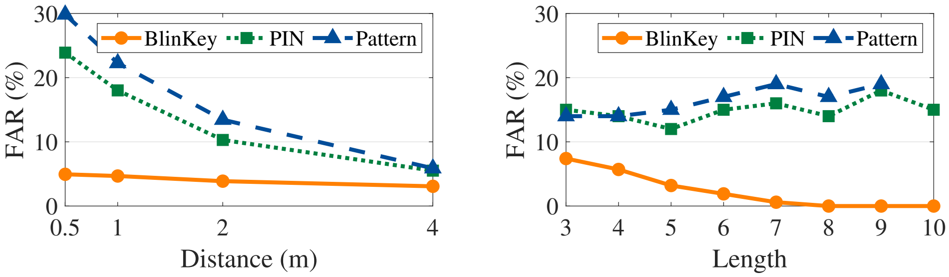
(h) #8 (i) #9 (j) #10 (k) #11 (l) #12 (m) #13

Figure 14. Visualization of the top-13 frequently selected blinkey patterns.

6.2.3 Credential-Aware Attacks. A credential-aware attack is when the adversary has the full knowledge of the blinking rhythm of a blinkey. Therefore, it can extract all the knowledge-based features, including blink time instances, blink intervals, and relative intervals. To launch this type of attack, we provide the attacker all the above-mentioned information regarding victim blinkeys. As discussed in statistical attacks, it is unlikely for the adversary to reproduce the legitimate user's biometrics. Likewise, to launch credential-aware attacks against PIN and pattern lock, adversaries are informed with details of victim PINs and drawing patterns. Based on this information, the attacker tries to gain access to the system. Table 8 compares the success rate against three types of authentication schemes. While PIN and pattern are compromised, *BlinKey* effectively resists the attack. This is because *BlinKey* also involves biometric features, which are hard to mimic, in addition to credentials. Meanwhile, we also notice that the leakage of credentials does provide attackers advantage in compromising the system. For instance, given the length of 7, the attacker's success rate is 0 under zero-effort attacks, while it increases to 14.2% under credential-aware attacks. This result indicates that biometric features alone, i.e., pupil size variations, cannot deliver satisfactory security performance. Luckily, the success rate against *BlinKey* is merely 4.4% when the length is 10. Therefore, one viable solution to defend credential-aware attacks is to adopt a longer blinkey. As a note, the length of a pattern lock is defined by the number of points a user draws through. For instance, the length of a "Z" pattern (1-2-3-5-7-8-9) is 7.

6.2.4 Shoulder-Surfing Attacks. Shoulder-surfing attacks are another general type of attacks against an authentication system, in which the adversary obtains authentication information via visual observation. It is more severe towards PIN/password/pattern authentication on VR devices than regular personal devices. Because the victim's vision is blocked by the headset, they are unaware of the surrounding environment, including the presence of shoulder-surfing attackers. We randomly pick 22 out of 43 participants involved in the phase-II user study and group them into 11 pairs. Each of them was told to replay his/her partner's passcode. Firstly, one user of the pair acts as an attacker, the other as a legitimate user, and then the roles are exchanged. During the experiment, the legitimate user repeats the same passcode for three times with a pause in between. Then, the attacker watches the entire process and tries to reproduce it. Every attacker makes three access attempts. The attacker is considered a success in a shoulder surfing if any one of the five trials passes the authentication.

Figure 15(a) plots the FAR, i.e., attacker's success rate, of *BlinKey*, PIN, and pattern lock with respect to its distance to the legitimate user. When the distance is 0.5 m, the success rate toward PIN and pattern lock is 23.9% and 29.8%, respectively, while that toward *BlinKey* is merely 4.9%. This is intuitive, as a shorter distance enables the attacker to have a closer observation over the legitimate user's login. Thus, it has a better chance to correctly replay the knowledge-based secret. On the other hand, it is hard, if not impossible, for the attacker to observe the user's eyes in a VR headset. Besides, as *BlinKey* involves biometric features, it is extremely challenging for an attacker to repeat such information. It also explains why FAR keeps almost unchanged as the distance gets longer. Figure 15(b) shows the success rate of should-surfing attacks with respect to the length of blinkey, PIN, and drawing pattern. Again, *BlinKey* has the best performance among the three. When the length is 8, FAR of *BlinKey* is 0, i.e., no adversary successfully launches shoulder-surfing attacks, while the value for PIN and pattern lock is 14.1% and 17.0%, respectively. Interestingly, unlike *BlinKey*, PIN and pattern lock become more vulnerable to shoulder-surfing attacks with a larger length. One possible explanation is that a longer key provides the attacker more information about the relative button positions to better infer the keypad structure.



(a) Impact of the victim-attacker distance

(b) Impact of key length

Figure 15. Success rate of shoulder-surfing attackers against *BlinKey*, PIN, and pattern lock.

The phenomenon that the success rate of shoulder-surfing attacks is non-zero is attributed to two reasons. First, while it is hard to launch the shoulder-surfing attack, the attacker can still guess the secret, i.e., zero-effort attack, even without much insight. As shown in Table 5 in the paper, its success rate is 8.1% when a blinkey has a length of only 3. Second, while k-NN exhibits promising authentication accuracy, it is imperfect. As shown in Figure 10, the lowest EER (where FAR=FRR) exists at 4.0%. It indicates that there is still certain possibility that an illegitimate blinkey is wrongly classified as a legitimate one. On the other hand, a close observation over an user's login process does provide the attacker some marginal advantage. For example, a couple of volunteers tend to nod their heads subconsciously following the same rhythm as they blink. This advantage diminishes quickly as the attack-victim distance increases.

6.3 Usability

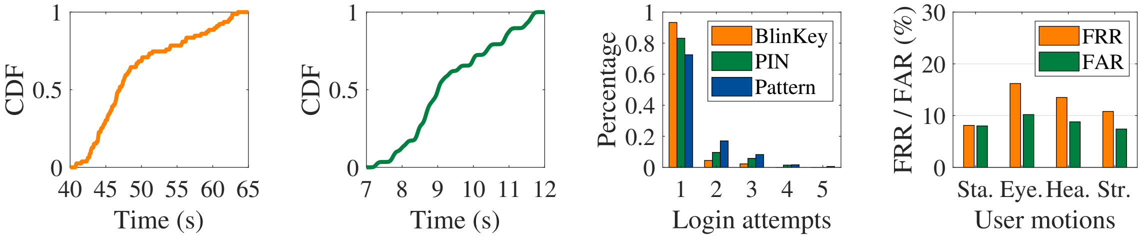
Apart from security, usability is another critical criterion to evaluate a user authentication scheme. We measure the usability of *BlinKey* from aspects of time consumption, legitimate recognition, memorability, and impact of user motions.

6.3.1 Time Consumption. We examine the enrollment time and login time needed for *BlinKey*. Specifically, the former refers to the total duration required to enroll all samples to train the classifier, while the latter is the total duration for a user to enter a test blinkey and for the system to make an authentication decision. The distributions of enrollment time and login time are depicted in Figure 16(a) and Figure 16(b), separately. We observe that the enrollment time of *BlinKey* ranges from 40.8 to 63.5 seconds. Its average, median, and 90-th percentile are 49.5 seconds, 42.9 seconds, and 61.1 seconds, respectively. The login time spans from 7.3 to 11.7 seconds, with its average, median, and 90-th percentile as 9.6 seconds, 8.9 seconds, and 11.2 seconds, respectively. Therefore, the most time-consuming part is the enrollment phase. Luckily, the enrollment only needs to be performed once for a user. Hence, its time consumption is still reasonably practical. The authentication time of our scheme is shorter than many existing solutions, such as [13, 62]. It takes 17 and 60 seconds to authenticate a user in [13] and [62], respectively. Besides, as shown in Table 4, only 4 out of 43 volunteers had the experience of performing authentication in a VR device before. This factor partially accounts for the time overhead in our result. We thus optimistically project that as users get more familiar with *BlinKey*, the enrollment and login time should be further reduced.

The blinks indicating the start and end of a blinkey have been taken into account for the measurement of both the enrollment time and login time in the evaluation. Specifically, 5 seconds out of the login duration (with the 90-th percentile as 11.2 seconds) are attributed to this overhead. As our future work, we plan to propose efficient approach to indicate the start/end of a blinkey with reduced overhead.

6.3.2 Login Attempts. This metric measures how many login attempts a legitimate user needs to unlock the device. A fewer number of attempts are desirable for an authentication scheme with high usability.

93.3% of blinkeys can be successfully authenticated in the first attempt, while this value for PIN and pattern lock is 83.2% and 72.5%, respectively. This is because users make mistakes more often in selecting the correct key or drawing the correct line on a virtual keyboard with the controllers. In contrast, the entering of blinkeys is performed by blinking eyes without interacting with the controller. It only takes 1.09 attempts on average for a legitimate user to get authenticated in *BlinKey*.



(a) Distribution of enrollment time (b) Distribution of login time (c) Login attempts (d) Impact of user motions

Figure 16. Evaluation of usability of *BlinKey*.

6.3.3 Memorability. Memorability demonstrates how well a secret key can be remembered by its owner, especially after a long period. To evaluate the memorability of *BlinKey*, we designed two follow-up experiments. The participants are invited to perform their *blinkey* after 7 days, and 14 days and test if they can successfully get authenticated. Out of the 45 volunteers who joined in the first-stage experiment, 29 and 15 of them participated in the two second-stage experiments, respectively. As shown in Table 9, 26 out of 29 volunteers are able to recall their blinkeys successfully after 7 days and 12 out of 15 volunteers are able to recall their blinkeys after 14 days. While the memorability performance of *BlinKey* is far from perfect, we would like to note that most of the volunteers may not have the chance to practice their blinkeys during 7 days, unlike regular passwords or

Table 9. The recall rate after a period of time.

Duration between stage-I and -II experiments	No. of participants	No. of successes	Success rate
After 7 days	29	26	89.6%
After 14 days	15	12	80.0%

digit-PINs that are entered to personal devices multiple times a day. We believe the performance will be enhanced with more frequent practices.

6.3.4 Impact of User Motions. In practical scenarios, users are not always sitting statically while entering a blinkey. Rather, they may be rotating their eyes, moving their heads, or even walking. An ideal system should be capable of handling these situations. In the experiment, we investigate whether user motions impact the performance of *BlinKey*. Four different types of motions are considered, sitting, rotating eyes, moving head, and strolling. We observe in Figure 16(d) that the best accuracy is achieved when the user is sitting, with its FRR at 8.1% and the FAR at 8.0%. The lowest accuracy is observed when the user is rotating eyes, with the corresponding FRR at 16.9% and FAR at 10.2%. This is because eye movement prevents the eye tracker to accurately estimate real-time pupil size. Nonetheless, neither head movement nor strolling causes significant performance degradation. Besides, we also observe that FAR is relatively stable across all motion status. It means the authentication security is not deteriorated much by motions. Based on the above observation, users will be recommended to enter blinkeys by looking into the virtual screen to prevent significant eyeball movement. There will be no restriction on their body movement, though.

6.4 Survey Results

In addition to the experiments, we further evaluate *BlinKey* via survey. The pre-survey was conducted after the introduction of the basic idea of *BlinKey* and before the experiment, while the post-survey is conducted after all experiments. Volunteers are asked to rate *BlinKey* from the perspectives of security and usability and compare them with commonly used methods on mobile devices, including PIN, password, and pattern lock. Questions include 1) Is it safe against attacks being tested? 2) Is it easy to perform and remember? On a 10-point Likert scale (1 = strongly disagree; 10 = strongly agree), participants pick a point that they deem proper. Survey results are shown in Figure 17. Most volunteers agree that *BlinKey* is better than the other three listed authentication methods in both aspects. It is worth mentioning that many participants rate *BlinKey* a higher score in the post-study than in the pre-study, which suggests that our scheme outperforms user's expectations.

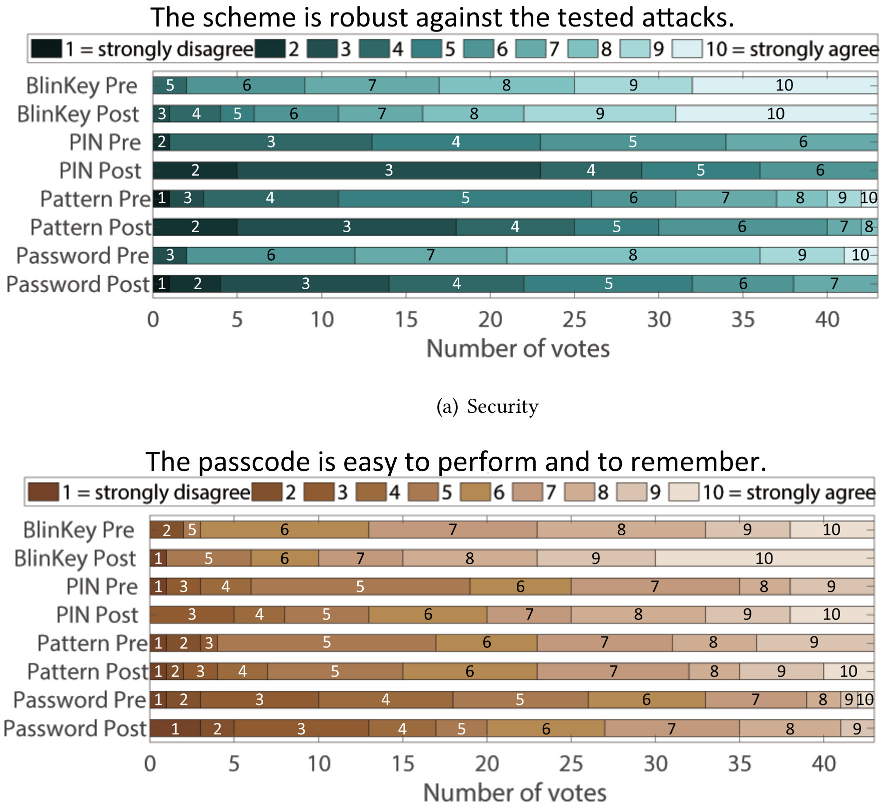
7 DISCUSSIONS

Raw size of *BlinKey* space. *BlinKey* is a two-factor authentication, a combination of the rhythm passcode and human biometrics, i.e., variations of pupil size. Since the variability brought by biometric features is hard to quantify, we would like to discuss the key space of *BlinKey* merely taking into account the variability introduced by blinking rhythms. Thus, the real key space of *BlinKey* should be no less than this value.

BlinKey adopts a similar design of the rhythm passcode as a prior work [36]. We thus revise the theoretical result of [36] and derive the key space of *BlinKey*.

Theorem 7.1. *(Revised from Theorem 5.1 of [36].) The size of* BlinKey*'s key space is*

$$|\Pi| = \sum_{l=1}^{L_{\max}} \binom{\frac{T_{\max}}{\sigma} - (\frac{\tau_b}{\sigma} - 1) \times l - (\frac{\tau_s}{\sigma} - 1) \times (l-1)}{2l-1},$$



(a) Security

(b) Usability

Figure 17. Pre-/Post-study survey results regarding security and usability.

where $L_{\max}$, $T_{\max}$, σ, τ_b and τ_s stand for the maximum blinkey length, corresponding maximum time duration, the system clock unit, minimum value of an onset-offset duration and minimum value of a offset-onset duration, respectively.

For an illustration purpose, we let $\sigma = 5$ ms, which is the time unit for Pupil Labs eye tracker's system clock. According to the statistic analysis over our collected dataset, we set the rest parameters as $T_{\max} = 12$ s, $\tau_s = 0.15$ s and $\tau_b = 0.10$ s. Thus, when the blinkey length is 6, the corresponding space size is about 10^{23}. As a reference, the key space for a regular PIN with 6 digits is 10^6. The above theorem is derived without considering pupil size variation. With the introduction of an additional dimension of entropy, the key space of *BlinKey* should be further enlarged.

Practical design. Our design grants the user some error tolerance–when a legitimate user fails to authenticate, she can re-enter her blinkey until the maximum number of attempts is reached. In this case, the user is temporarily locked out, and the recovery process is invoked (see Figure 12(f)). Here are two classic recovery methods widely adopted by other user authentication schemes. 1) Provide an alternative way to authenticate users; when a legitimate user fails to authenticate herself with her blinkey, she can still unlock the device by entering a valid passcode or digit-PIN. 2) Have a remote server to send a recovery code to the user's previously authorized email address; the user retrieves the code by accessing the email and unlocks the device by entering the code. These two approaches are deemed robust against attacks.

When an adversary tends to enroll himself in the device, he needs to first enter a valid blinkey, which has been created by the legitimate user earlier, to unlock the device. Otherwise, there is no way for the adversary to enroll himself. This idea has been adopted in many personal devices, such as smartphones and PCs. There is also an exception that the victim VR device has not been secured with any user authentication scheme. In this case, the adversary can directly set up his account associated with his blinkey in the device. To address this issue, a conventional solution is to enforce the user to enroll her authentication credentials, i.e., blinkey here, during initial account setup.

Impact of environment. User's pupil size is affected by their biophysical status, such as mood, energy level, whether drinking alcohol, illness, etc. Consequently, these factors would impair the authentication accuracy of *BlinKey*. One viable solution is to further deploy a second-option user authentication method, such as digit-PIN or password. Once a legitimate user's input cannot be recognized by the system by any chance, including the above-mentioned situations, she can always unlock the device by a valid digit-PIN. Such an idea has been adopted by current fingerprint-/facial recognition-based user authentication on smartphones. While the brightness of the display does affect pupil size when blinking, it does not necessarily impact the performance of our scheme. As shown in Figure 12(c), the screen displays the same image with the same brightness/color/content during the login process. Thus, it eliminates the impact from the display.

Reduce login overhead. Under the current design, the login duration of *BlinKey* spans from 7.3 to 11.7 seconds, with its average, median, and 90-th percentile as 9.6 seconds, 8.9 seconds, and 11.2 seconds, respectively. While this overhead is reasonably practical, it is still longer than conventional PIN and password. The most significant portion of the overhead is attributed to the blinks indicating the start and end of a blinkey, i.e., 5 seconds according to the setting. As our future work, we plan to propose efficient approach to indicate the start/end of a blinkey with reduced overhead. Besides, as shown in Table 4, only 4 out of 43 volunteers had the experience of performing authentication in a VR device before. This factor partially accounts for the long time overhead in our result. The login time would be further reduced as users get more familiar with authenticating themselves via *BlinKey* in VR.

8 CONCLUSIONS

As VR devices are increasingly weaved into our everyday life, providing security to the data acquired by or accessed through these devices becomes critically important. In this study, we develop a two-factor user authentication mechanism, named *BlinKey*, which employs the user-designed blinking rhythm and unique biometrics exhibited in pupil size variation to fingerprint legitimate users. Compared to prior work, our solution delivers secure authentication, incurs low cognitive overhead, and offers great convenience. Through an extensive evaluation that involves 52 volunteers, we observe that the average EER is as low as 4.0% with only 6 training samples. The proposed *BlinKey* is also implemented on an HTC Vive Pro with a Pupil Labs eye tracker. We further measure its security by testing robustness against various types of attackers, and its utility, from aspects of time consumption, login attempts, the impact of user motions, and memorability. We observe that *BlinKey* requires relatively long enrollment time (median: 42.9 seconds). One reason is that many participants have limited experience in authenticating themselves on VR devices. This is likely to be alleviated as users practice it multiple times daily after scheme implementation. Besides, as enrollment is only executed once for each blinkey, the long enrollment time will not incur noticeable overhead from a long-term view. In conclusion, we believe *BlinKey* is a practical authentication method applicable to current VR devices.

ACKNOWLEDGMENTS

We sincerely thank the anonymous reviewers for their insightful comments and suggestions. We are also grateful to NSF (CNS-1943509, ECCS-1849860) for partially funding this research.

REFERENCES

[1] Viar 360. 2017. Virtual reality in education – how are schools using VR? https://www.viar360.com/education-schools-using-virtual-reality/

[2] Michael Abehsera. 2020. 3 ways virtual reality will transform e-commerce. https://www.toptal.com/insights/innovation/3-ways-virtual-reality-transforms-ecommerce

[3] Mayank Agarwal, Mahendra Mehra, Renuka Pawar, and Deven Shah. 2011. Secure authentication using dynamic virtual keyboard layout. In *Proceedings of the International Conference Workshop on Emerging Trends in Technology (ICWET '11)*. Association for Computing Machinery, 288–291.

[4] Fawaz Alsulaiman and Abdulmotale El Saddik. 2006. A novel 3D graphical password schema. In *Proceedings of the 2006 IEEE Symposium on Virtual Environments, Human-Computer Interfaces and Measurement Systems*. 125–128.

[5] Cemil Altin and Orhan Er. 2016. Comparison of different time and frequency domain feature extraction methods on elbow gesture's EMG. *European Journal of Interdisciplinary Studies* 5 (August 2016), 35.

[6] Ilhan Aslan, Andreas Uhl, Alexander Meschtscherjakov, and Manfred Tscheligi. 2014. Mid-air authentication gestures: An exploration of authentication based on palm and finger motions. In *Proceedings of the 16th International Conference on Multimodal Interaction (ICMI '14)*. Association for Computing Machinery, 311–318.

[7] Nick Babich. 2019. How VR in education will change how we learn and teach. https://xd.adobe.com/ideas/principles/emerging-technology/virtual-reality-will-change-learn-teach/

[8] Roman Bednarik, Tomi Kinnunen, Andrei Mihaila, and Pasi Fränti. 2005. Eye-movements as a biometric. In *Scandinavian conference on image analysis*. Springer, 780–789.

[9] Ariel Bogle. 2020. eBay launches a world-first virtual reality department Store. http://mashable.com/2016/05/18/ebay-virtual-reality-shopping/#MqZVNlqvUEqf.

[10] Bhavana Borkar, Shiba Sheikh, and PD Kaware. 2016. 4D password mechanism. In *Imperial Journal of Interdisciplinary Research*, Vol. 2. 240–245.

[11] Steven Brand. 2020. How virtual reality is changing the manufacturing game. https://www.cmtc.com/blog/how-virtual-reality-is-changing-the-manufacturing-game

[12] Davina Bristow, John-Dylan Haynes, Richard Sylvester, Christopher D. Frith, and Geraint Rees. 2005. Blinking Suppresses the Neural Response to Unchanging Retinal Stimulation. *Current Biology* 15, 14 (June 2005), 1296 – 1300. https://doi.org/10.1016/j.cub.2005.06.025

[13] Virginio Cantoni, Chiara Galdi, Michele Nappi, Marco Porta, and Daniel Riccio. 2015. GANT: Gaze analysis technique for human identification. *Pattern Recognition* 48 (2015), 1027–1038.

[14] Supply Chain Game Changer. 2020. Virtual reality (VR) is enhancing e-commerce shopping! https://supplychaingamechanger.com/how-virtual-reality-vr-is-drastically-enhancing-the-e-commerce-shopping-experience-infographic/

[15] Yimin Chen, Jingchao Sun, Rui Zhang, and Yanchao Zhang. 2015. Your song your way: Rhythm-based two-factor authentication for multi-touch mobile devices. In *Proceedings of the 2015 IEEE Conference on Computer Communications*. 2686–2694.

[16] Jennifer Clopton. 2020. Virtual reality brings new vision to health care. https://www.webmd.com/cancer/news/20200210/virtual-reality-brings-new-vision-to-health-care

[17] Sauvik Das, Gierad Laput, Chris Harrison, and Jason I Hong. 2017. Thumprint: Socially-inclusive local group authentication through shared secret knocks. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, 3764–3774.

[18] Statista Research Department. 2020. Global virtual reality device shipments by vendor. https://www.statista.com/statistics/671403/global-virtual-reality-device-shipments-by-vendor/

[19] Chesner Désir, Simon Bernard, Caroline Petitjean, and Heutte Laurent. 2013. One class random forests. *Pattern Recognition* 46, 12 (2013), 3490–3506. https://doi.org/10.1016/j.patcog.2013.05.022

[20] Simon Eberz, Kasper Bonne Rasmussen, Vincent Lenders, and Ivan Martinovic. 2015. Preventing lunchtime attacks: Fighting insider threats with eye movement biometrics. In *The Network and Distributed System Security Symposium*.

[21] Malin Eiband, Mohamed Khamis, Emanuel von Zezschwitz, Heinrich Hussmann, and Florian Alt. 2017. Understanding shoulder surfing in the wild: Stories from users and observers. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, 4254–4265.

[22] eMarkerter. 2020. US Virtual and Augmented Reality Users 2020. https://www.emarketer.com/content/us-virtual-and-augmented-reality-users-2020/

[23] Wells Fargo. 2020. Biometric Authentication. https://www.wellsfargo.com/online-banking/biometric/

[24] Michael Fauscette. 2020. Biometrics are coming so are security concerns. https://www.darkreading.com/endpoint/biometrics-are-coming-and-so-are-security-concerns/a/d-id/1331536

[25] Caleb Finch. 2018. Manufacturing with VR becoming a (virtual) reality. https://blog.qad.com/2018/09/manufacturing-vr-becoming-virtual-reality/

[26] FOVE. 2016. FOVE 0 eye tracking VR devkit for developers, creators, researchers. https://www.getfove.com/

[27] Markus Funk, Karola Marky, Iori Mizutani, Mareike Kritzler, Simon Mayer, and Florian Michahelles. 2019. LookUnlock: Using spatial-targets for user-authentication on HMDs. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems (CHI EA '19)*. Association for Computing Machinery, Article Paper LBW0114, 6 pages.

[28] Albizu Garcia. 2019. Is virtual reality the future of social networking? https://sociable.co/social-media/is-virtual-reality-future-social-networking/

[29] Ceenu George, Mohamed Khamis, Daniel Buschek, and Heinrich Hussmann. 2019. Investigating the third dimension for authentication in immersive virtual reality and in the real world. In *Proceedings of the 2019 IEEE Conference on Virtual Reality and 3D User Interfaces*. 277–285.

[30] Ceenu George, Mohamed Khamis, Emanuel von Zezschwitz, Marinus Burger, Henri Schmidt, Florian Alt, and Heinrich Hußmann. 2017. Seamless and secure VR: Adapting and evaluating established authentication systems for virtual reality. In *Network and Distributed System Security Symposium*.

[31] Stephen Gossett. 2020. Virtual reality in education: An overview. https://builtin.com/edtech/virtual-reality-in-education

[32] Kelly S Hale and Kay M Stanney. 2014. *Handbook of virtual environments: Design, implementation, and applications*. CRC Press.

[33] Corey D Holland and Oleg V Komogortsev. 2011. Biometric identification via eye movement scanpaths in reading. In *2011 International Joint Conference on Biometrics*. 1–8.

[34] Corey D Holland and Oleg V Komogortsev. 2013. Complex eye movement pattern biometrics: Analyzing fixations and saccades. *Proceedings of the 2013 International Conference on Biometrics, ICB 2013*, 1–8. https://doi.org/10.1109/ICB.2013.6612953

[35] HTC. 2020. HTC Vive Pro Eye. https://www.vive.com/eu/product/vive-pro-eye/

[36] Ben Hutchins, Anudeep Reddy, Wenqiang Jin, Michael Zhou, Ming Li, and Lei Yang. 2018. Beat-PIN: A user authentication mechanism for wearable devices through secret beats. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security (ASIACCS '18)*. Association for Computing Machinery, 101–115.

[37] Andrew Hutchinson. 2020. Facebook begins user testing of new 'Horizon' VR social platform. https://www.socialmediatoday.com/news/facebook-begins-user-testing-of-new-horizon-vr-social-platform/573852/

[38] Mordor Intelligence. 2020. Virtual reality (VR) market - growth, trends, and forecast (2020 - 2025). https://www.mordorintelligence.com/industry-reports/virtual-reality-market

[39] Kapil Jain and Nirbhay Pherwani. 2017. Virtual reality based user authentication system. In *International Journal of Science Technology Engineering*, Vol. 4. 49–53.

[40] Tomi Kinnunen, Filip Sedlak, and Roman Bednarik. 2010. Towards task-independent person authentication using eye movement signals. In *Proceedings of the 2010 Symposium on Eye-Tracking Research Applications (ETRA '10)*. Association for Computing Machinery, 187–190.

[41] Nadia Kovics. 2020. Virtual reality in military. https://thinkmobiles.com/blog/virtual-reality-military/

[42] Alexander Kupin, Benjamin Moeller, Yijun Jiang, Natasha Kholgade Banerjee, and Sean Banerjee. 2019. *Task-driven biometric authentication of users in virtual reality (VR) environments: 25th International Conference, MMM 2019, Thessaloniki, Greece, January 8–11, 2019, Proceedings, Part I*. 55–67.

[43] Oscar D Lara and Miguel A Labrador. 2013. A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys Tutorials* 15, 3 (March 2013), 1192–1209.

[44] Feng Lin, Kun Woo Cho, Chen Song, Wenyao Xu, and Zhanpeng Jin. 2018. Brain password: A secure and truly cancelable brain biometrics for smart headwear. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '18)*. Association for Computing Machinery, 296–309.

[45] Jian Liu, Chen Wang, Yingying Chen, and Nitesh Saxena. 2017. VibWrite: Towards finger-input authentication on ubiquitous surfaces via physical vibration. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS '17)*. Association for Computing Machinery, 73–87.

[46] Dong Ma, Guohao Lan, Mahbub Hassan, Wen Hu, Mushfika B Upama, Ashraf Uddin, and Moustafa Youssef. 2019. SolarGest: Ubiquitous and battery-free gesture recognition using solar cells. In *Proceedings of the 25th Annual International Conference on Mobile Computing and Networking (MobiCom '19)*. Association for Computing Machinery, Article Article 12, 15 pages.

[47] Florian Mathis, Hassan Ismail Fawaz, and Mohamed Khamis. 2020. Knowledge-Driven Biometric Authentication in Virtual Reality. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (CHI EA '20)*. Association for Computing Machinery, 1–10. https://doi.org/10.1145/3334480.3382799

[48] Florian Mathis, John Williamson, Kami Vaniea, and Mohamed Khamis. 2020. RubikAuth: Fast and Secure Authentication in Virtual Reality. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (CHI EA '20)*. Association for Computing

Machinery, 1–9. https://doi.org/10.1145/3334480.3382827

[49] Michael Morozov. 2019. Virtual reality in manufacturing. https://jasoren.com/virtual-reality-manufacturing/

[50] Tahrima Mustafa, Richard Matovu, Abdul Serwadda, and Nicholas Muirhead. 2018. Unsure how to authenticate on your VR headset?: Come on, use your head!. In *Proceedings of the 4th ACM International Workshop on Security and Privacy Analytics (IWSPA '18)*. 23–30.

[51] Toan Nguyen and Nasir Memon. 2018. Tap-based user authentication for smartwatches. *Computers and Security* 78 (September 2018), 174–186. https://doi.org/10.1016/j.cose.2018.07.001

[52] Oculus. 2020. Facebook Horizon. https://www.oculus.com/facebookhorizon/?locale=en_US

[53] Bank of America. 2020. Access your account securely with fingerprint sign-in. https://www.bankofamerica.com/online-banking/mobile-and-online-banking-features/touch-id/

[54] Internet of Business. 2020. Alibaba launches VR Pay, gives virtual reality payments the nod. https://internetofbusiness.com/alibaba-vr-pay-virtual-reality/

[55] The Database of Useful Biological Numbers. 2001. Average duration of a single eye blink. https://bionumbers.hms.harvard.edu/bionumber.aspx?&id=100706&ver=4

[56] Poojan Oza and Vishal Patel. 2019. Active Authentication using an Autoencoder regularized CNN-based One-Class Classifier. 1–8. https://doi.org/10.1109/FG.2019.8756525

[57] P. Oza and V. M. Patel. 2019. One-Class Convolutional Neural Network. *IEEE Signal Processing Letters* 26, 2 (2019), 277–281.

[58] Ken Pfeuffer, Matthias Geiger Geiger, Sarah Prange, Lukas Mecke, Daniel Buschek, and Florian Alt. 2019. Behavioural biometrics in VR: Identifying people from body motion and relations in virtual reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, Article Paper 110, 12 pages.

[59] Pico. 2020. Pico Neo 2. https://www.pico-interactive.com/us/neo2.html

[60] Sundaramurthi Rajarajan, K Kavitha Maheswari, R Hemapriya, and S Sriharilakshmi. 2014. Shoulder surfing resistant virtual keyboard for internet banking. *World Applied Sciences Journal* 31, 7 (2014), 1297–1304.

[61] Ioannis Rigas, George Economou, and Spiros Fotopoulos. 2012. Biometric identification based on the eye movements and graph matching techniques. *Pattern Recognition Letters* 33 (2012), 786–792.

[62] Ioannis Rigas and Oleg Komogortsev. 2014. Biometric recognition via probabilistic spatial projection of eye movement trajectories in dynamic visual environments. *Information Forensics and Security, IEEE Transactions on* 9 (2014), 1743–1754.

[63] Giuseppe Riva and Brenda K Wiederhold. 2015. *The new dawn of virtual reality in health care: Medical simulation and experiential interface.* SHTI '15, Vol. 219. IOS Press. 3–6 pages.

[64] Sol Rogers. 2019. Seven reasons why eye-tracking will fundamentally change VR. https://www.forbes.com/sites/solrogers/2019/02/05/seven-reasons-why-eye-tracking-will-fundamentally-change-vr/#22e0ef2c3459

[65] Kenneth S Saladin. 2012. *Anatomy and physiology.* McGraw-Hill.

[66] Stefan Schneegass, Youssef Oualil, and Andreas Bulling. 2016. SkullConduct: Biometric User Identification on Eyewear Computers Using Bone Conduction Through the Skull. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, 1379–1384. https://doi.org/10.1145/2858036.2858152

[67] Ivo Sluganovic, Marc Roeschlin, Kasper B Rasmussen, and Ivan Martinovic. 2016. Using reflexive eye movements for fast challenge-response authentication. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16)*. Association for Computing Machinery, 1056–1067.

[68] Virtual Reality Society. 2017. Virtual reality in the military. https://www.vrs.org.uk/virtual-reality-military/

[69] Yunpeng Song, Zhongmin Cai, and Zhi-Li Zhang. 2017. Multi-touch authentication using hand geometry and behavioral information. In *Proceedings of the 2017 IEEE Symposium on Security and Privacy*. 357–372.

[70] Scott Stein. 2020. Eye tracking is the next phase for VR, ready or not. https://www.cnet.com/news/eye-tracking-is-the-next-phase-for-vr-ready-or-not/

[71] Qbit Technologies. 2020. How VR will revolutionize e-commerce. https://www.qbittech.com/index.php/vr-blog/item/130-the-future-of-e-commerce-is-virtual-reality

[72] Varjo. 2020. Varjo VR-1: The first human-eye resolution headset. https://varjo.com/products/vr-1/

[73] Michael Velichko. 2019. VR military training – the next step of combat evolution. https://jasoren.com/vr-military-training-the-next-step-of-combat-evolution/

[74] Visualise. 2020. Virtual reality in healthcare. https://visualise.com/virtual-reality/virtual-reality-healthcare

[75] Tracy Watson. 2019. VR social media: Is it the future of social interaction? https://skywell.software/blog/vr-social-media-future/

[76] Wikipedia. 2020. Blinking. https://en.wikipedia.org/wiki/Blinking

[77] Wikipedia. 2020. Spline interpolatoin. https://en.wikipedia.org/wiki/Spline_interpolation

[78] Jacob Otto Wobbrock. 2009. TapSongs: Tapping rhythm-based passwords on a single binary sensor. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. Association for Computing Machinery, 93–96.

[79] Zhen Yu, Hai-Ning Liang, Charles Fleming, and Ka Lok Man. 2016. An exploration of usable authentication mechanisms for virtual reality systems. In *Proceedings of the 2016 IEEE Asia Pacific Conference on Circuits and Systems*. 458–460.

[80] Mei Zhang, Jinglan Wu, Huifeng Lin, Peng Yuan, and Yanan Song. 2017. The Application of One-Class Classifier Based on CNN in Image Defect Detection. *Procedia Computer Science* 114 (2017), 341 – 348. https://doi.org/10.1016/j.procs.2017.09.040 Complex Adaptive Systems Conference with Theme: Engineering Cyber Physical Systems, CAS October 30 – November 1, 2017, Chicago, Illinois, USA.

[81] Yongtuo Zhang, Wen Hu, Weitao Xu, Chun Tung Chou, and Jiankun Hu. 2018. Continuous authentication using eye movement response of implicit visual stimuli. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4, Article Article 177 (January 2018), 22 pages.