Camera Based Two Factor Authentication Through Mobile and Wearable Devices

MOZHGAN AZIMPOURKIVI, Florida International University
UMUT TOPKARA, Bloomberg LP
BOGDAN CARBUNAR, Florida International University

We introduce Pixie, a novel, camera based two factor authentication solution for mobile and wearable devices. A quick and familiar user action of snapping a photo is sufficient for Pixie to simultaneously perform a graphical password authentication and a physical token based authentication, yet it does not require any expensive, uncommon hardware. Pixie establishes trust based on both the knowledge and possession of an arbitrary physical object readily accessible to the user, called *trinket*. Users choose their trinkets similar to setting a password, and authenticate by presenting the same trinket to the camera. The fact that the object is the trinket, is secret to the user. Pixie extracts robust, novel features from trinket images, and leverages a supervised learning classifier to effectively address inconsistencies between images of the same trinket captured in different circumstances.

Pixie achieved a false accept rate below 0.09% in a brute force attack with 14.3 million authentication attempts, generated with 40,000 trinket images that we captured and collected from public datasets. We identify *master* images, that match multiple trinkets, and study techniques to reduce their impact.

In a user study with 42 participants over 8 days in 3 sessions we found that Pixie outperforms text based passwords on memorability, speed, and user preference. Furthermore, Pixie was easily discoverable by new users and accurate under field use. Users were able to remember their trinkets 2 and 7 days after registering them, without any practice between the 3 test dates.

CCS Concepts: • **Security and privacy** → **Authentication**; *Usability in security and privacy*;

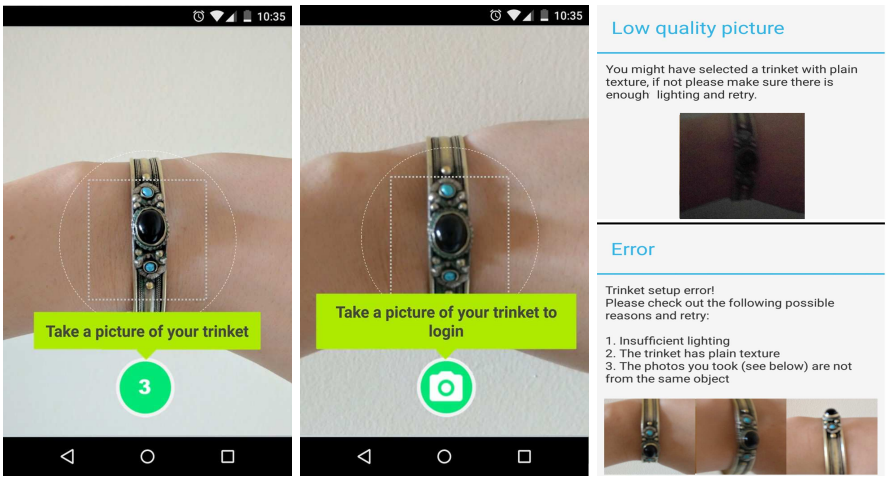
Additional Key Words and Phrases: Multi-factor authentication, Mobile and wearable device authentication

ACM Reference format:
Mozhgan Azimpourkivi, Umut Topkara, and Bogdan Carbunar. 2017. Camera Based Two Factor Authentication Through Mobile and Wearable Devices. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article 35 (September 2017), 37 pages.
DOI: 10.1145/3130900

1 INTRODUCTION

Mobile and wearable devices are popular platforms for accessing sensitive online services such as e-mail, social networks and banking. A secure and practical experience for user authentication in such devices is challenging, as their small form factor, especially for wearables (e.g., smartwatches [63] and smartglasses [80]), complicates the input of commonly used text based passwords, even when the memorability of passwords already poses a significant burden for users trying to access a multitude of services [14]. While the small form factor of mobile and wearable devices makes biometric authentication solutions

DOI: 10.1145/3130900



(a) (b) (c)

Fig. 1. Pixie: (a) Trinket setup. The user takes photos of the trinket placing it in the circle overlay. UI shows the number of photos left to take. (b) Login: the user snaps a photo of the trinket. (c) Trinket setup messages provide actionable guidance, when the image quality is low (top), or the reference images are inconsistent (bottom).

seemingly ideal, their reliance on sensitive, hard to change user information introduces important privacy and security issues [55, 56] of massive scale.

In this paper we introduce Pixie, a camera based remote authentication solution for mobile devices, see Figure 1 and [68] for a short demo. Pixie can establish trust to a remote service based on the user's ability to present to the camera a previously agreed secret physical token. We call this token, the *trinket*. We use the term trinket to signify the uniqueness and small size of the token, not its value.

Just like setting a password, the user picks a readily accessible trinket of his preference, e.g., a clothing accessory, a book, or a desk toy, then uses the device camera to snap trinket images (a.k.a., *reference* images). All the user needs to do to authenticate is to point the camera to the trinket. If the captured *candidate* image matches the reference images, the authentication succeeds.

Pixie combines graphical password [7, 19, 53] and token based authentication concepts [59, 79], into a two factor authentication (2FA) solution based on what the user has (the trinket) and what the user knows - the trinket, the angle and section used to authenticate. Figure 2 shows examples of trinkets. Contrary to other token based authentication methods, Pixie does not require expensive, uncommon hardware to act as the second factor; that duty is assigned to the physical trinket, and the mobile device in Pixie is the primary device through which the user authenticates. Pixie only requires the authentication device to have a camera, making authentication convenient even for wearable devices such as smartwatches and smartglasses.

Challenges and proposed approach. Building a secure and usable trinket based authentication solution is difficult. Unlike biometrics based solutions, trinkets can be chosen from a more diverse space than e.g., faces, thus lack the convenience of a set of well known features. In addition, users cannot be expected to accurately replicate during login, the conditions (e.g. angle, distance and background) of the trinket setup process. Thus, Pixie needs to be resilient to candidate images captured in different circumstances than the reference images. Pixie addresses these problems in two ways: i) during the registration phase users are asked to capture multiple trinket images, thereby revealing the variability of the trinket to Pixie, ii) to match a candidate image against these reference images, Pixie leverages

(a) (b) (c) (d) (e) (f)

Fig. 2. Examples of good (a-c) and low quality (d-f) trinket images. Trinkets are small (parts of) objects carried or worn by users, thus hard to steal and even reproduce by adversaries. ORB keypoints are shown as small, colored circles. Good images have a high number of keypoints on the trinket. Low quality images are due to (d) insufficient light conditions on shirt section, (e) bright light and reflection, (f) image blur, or uniform, texture-less trinket.

a statistical classifier using features which leverage robust keypoints [3, 60] extracted from the trinket images.

In addition, in early pilot user studies, we identified new challenges for a successful deployment of Pixie. First, that Pixie users may use low quality trinkets, e.g. with uniform textures, capture inconsistent reference images with largely different viewing angles, or capture low quality images of their trinkets, e.g., blurry, or with improper lighting conditions, see Figure 2(d)-(f). In order to help the users pick high quality trinkets and images thereof, we develop features that capture the quality of reference images as defined by the likelihood of causing false accepts or false rejects during authentication. We use these features to train a trinket image rejection classifier that detects low quality images before they can be used as Pixie trinkets.

Second, we found that it is crucial to give the user actionable feedback about how to choose a better trinket when the Pixie filter rejects trinket images. For instance, a set of reference images can be rejected because they contain different trinkets, or because one of the images is blurry. However, most statistical classifiers are not easily interpretable, thus cannot indicate the nature of the problem. In order to provide meaningful actionable feedback, we identify feature threshold values that pinpoint problem images and naturally translate them into user instructions (see Table 5).

Implementation and evaluation. We implement Pixie for Android, and show using an extensive evaluation that Pixie is secure, fast, and usable. Pixie achieves a False Accept Rate (FAR) of 0.02% and a False Reject Rate (FRR) of 4.25%, when evaluated over 122,500 authentication instances. Pixie processes a login attempt in 0.5s on a HTC One (2013 Model, 1.7GHz CPU, 2GB RAM).

To evaluate the security of Pixie, we introduce several image based attacks, including an image based dictionary (or "pictionary") attack. Pixie achieves a FAR below 0.09% on such an attack consisting of 14.3 million authentication attempts constructed using public trinket image datasets and images that we collected online. Similar to face based authentication, Pixie is vulnerable to attacks where the adversary captures a picture of the trinket. However, we show that Pixie is resilient to a shoulder surfing attack flavor where the adversary knows or guesses the victim's trinket object type. Specifically, on a targeted attack dataset of 7,853 images, the average number of "trials until success" exceeds 5,500 irrespective of whether the adversary knows the trinket type or not. In addition, we introduce and study the concept of *master* images, whose diverse keypoints enable them to match multiple trinkets. We develop features that enable Pixie to reduce the effectiveness of master images.

We perform a user study with 42 participants over 8 days in 3 sessions, and show that Pixie is discoverable: *without prior training* and given no external help, 86% and 78% of the participants were

able to correctly set a trinket then authenticate with it, respectively. Pixie's trinkets were perceived as more memorable than text passwords, and were also easily remembered 2 and 7 days after being set.

Further, without any additional practice outside of the 3 sessions, participants entered their trinket progressively faster than their text passwords. Participants believed that Pixie is easier to use, more memorable and faster than text passwords. We found that the preference of Pixie over text passwords correlates positively with its preference on ease of use, memorability and security dimensions and overall perception of trinket memorability and willingness to adopt Pixie. In addition, 50% of participants reported that they preferred Pixie over text passwords.

In summary, we introduce the following contributions:

- **Pixie**. We introduce Pixie, a two factor, mobile device based authentication solution, that leverages the ubiquitous cameras of mobile devices to snap images of trinkets carried by the users. Pixie makes mobile device based authentication fast and convenient, and does not require expensive, uncommon hardware. Pixie leverages a novel set of features that determine if a candidate image contains the same token as a set of reference images [§ 4.3]. We develop filters that identify low quality images and inconsistent reference images, and provide actionable feedback to the users [§ 4.4].
- **Security**. We develop several image based attacks including brute force image dictionary attacks, a shoulder surfing flavor and master image attacks. We construct more than 14.3 million authentication instances to show that Pixie is resilient to these attacks [§ 5.3].
- **User study**. We implement Pixie in Android, and show through a user study with 42 participants that it is accurate, faster than text passwords, perceived as such by users, and its trinkets are memorable [§ 5].
- **Reproducibility**. Pixie is an open source prototype, with code and the Android installation file available on GitHub [12] and the Google Play Store [10]. We have also made our datasets, including the Pixie attack datasets, available for download [11].

2 RELATED WORK

Pixie is a camera based authentication solution that combines graphical password and token based authentication concepts, into a single step 2 Factor Authentication (2FA) solution. Pixie authentication is based on what the user has (the trinket) and what the user knows (the particular trinket among all the other objects that the user readily has access to, angle and viewpoint used to register the trinket). The unique form factor of Pixie differentiates it from existing solutions based on typed, drawn, or spoken secrets. We briefly survey and distinguish Pixie from existing solutions.

2.1 Mobile Biometrics

Biometric based mobile authentication solutions leverage unique human characteristics, e.g., faces [21], fingerprints [2], gait [38], to authenticate users. In particular, the Pixie form factor makes it similar to camera based biometric authentication solutions based on face [8, 21, 76] and gaze [40, 44]. Consequently, Pixie shares several limitations with these solutions, that include (i) vulnerability to shoulder surfing attacks and (i) susceptibility to inappropriate lighting conditions, that can spoil the performance and usability of the authentication mechanism [4, 45].

In contrast to biometrics, Pixie enables users to change the authenticating physical factor, as they change accessories they wear or carry. This reduces the risks from an adversary who has acquired the authentication secret from having lifelong consequences for the victims, thereby mitigating the need for biometric traceability and revocation [56].

Table 1. Comparison of usability related metrics of Pixie's camera based two-factor authentication approach with text, biometric and graphical password authentication solutions. **The Pixie user entry time is faster than typing text passwords**. The results of text-based passwords evaluated in § 6.2 are consistent with those from previous work. Pixie's median of login trials until success is 1, similar to other solutions.

Solution	Success rate (%)	Entry Time (s)	Number of trials before success
Pixie	**84.00**	**7.99 (Std=2.26, Mdn=8.51)**	**1.2 (Std=0.4, Mdn=1)**
Text password (MyFIU)	88.10	12.5 (Std=6.5, Mdn=11.5)	1.4 (Std=1.02, Mdn=1)
Text password (comp8) [70]*	75.0-80.1	(Mdn=13.2)	1.3
Eye tracking [44]	77.2-91.6	¡ 9.6	1.37 (Std=0.8, Mdn=1)-1.05 (Std=0.3, Mdn=1)
GazeTouchPass [40]	65	3.13	1.9 (Std=1.4, Mdn=1)
Face biometric [76]	96.9	(Mdn=5.55)	N/A
Face & eyes [8]*	N/A	20-40	1.1
Face & voice [76]	78.7	(Mdn=7.63)	N/A
Voice biometric [76]	99.5	(Mdn=5.15)	N/A
Gesture (stroke) biometric [76]	100	(Mdn=8.10)	N/A
Android pattern unlock [34]	87.92	0.9 (Std=0.63, Mdn=0.74)	1.13(Std=0.06, Mdn=1.11)
Passpoints [14]*	57	18.1 (Mdn=15.7)	2.2
Xside [22]	88	3.1-4.1	N/A
SmudgeSafe [66]	74	3.64 (Std=1.66)	N/A

* The study device is a computer.

Table 1 compares the user entry times of Pixie with various other authentication solutions. While Pixie takes longer than biometric authentication based on face [76], it is still faster than several authentication solutions based on gaze [8, 44]. We note that while fingerprint based authentication is fast and convenient [4], it is only applicable to devices that invest in such equipment. In contrast, cameras are ubiquitously present, including on wearable devices such as smartwatches and smartglasses.

Pixie needs to solve a harder problem than existing biometrics based authentication solutions, due to the diversity of its trinkets: while existing biometrics solutions focus on a single, well studied human characteristic, Pixie's trinkets can be arbitrary objects.

2.2 Security Tokens and 2 Factor Authentication (2FA)

The trinket concept is similar to hardware security tokens [59], as authentication involves access to a physical object. Hardware tokens are electronic devices that provide periodically changing one time passwords (OTP), which the user needs to manually enter to the authentication device. Mare et al. [46] found that 25% of authentications performed in the daily life employed physical tokens (e.g. car keys, ID badges, etc.).

Common software token solutions such as Google's 2-step verification [33], send a verification code to the mobile device, e.g. through SMS or e-mail. The user needs to retrieve the verification code (second authentication factor) and type it into the authentication device. This further requires the device to be reachable from the server hence introduces new challenges, e.g. location tracing, delays in phone network, poor network coverage. Moreover, such solutions provide no protection when the device is stolen. They also impact usability, as the user needs to type both a password and the verification code. In contrast, the Pixie trinket combines the user's secret and the second authentication factor. It also reduces user interaction, by replacing the typing of two strings with snapping a photo of the trinket.

Solutions such as [17, 39, 71] treat the mobile device as a second factor and eliminate user interaction to retrieve a token from the mobile device to the authentication device (e.g. a desktop) by leveraging

proximity based connectivity (e.g., Bluetooth, Wi-Fi). In contrast, Pixie assigns the duty of storing the token for the second factor to a physical object outside the mobile device. The mobile device is the sole device that is used to access the services on remote servers. As an added benefit, the physical factor of the trinket renders Pixie immune to the "2FA synchronization vulnerabilities" introduced by Konoth et al. [42], that exploit the ongoing integration of apps among multiple platforms.

Several authentication solutions rely on visual tokens (e.g., barcodes or QR codes) that are presented to the authentication device camera for verification [25, 35, 47, 71]. For instance, McCune et al. [47] use the camera phone as a visual channel to capture a 2D barcode, that encodes identifying cryptographic information (e.g., the public key of another device). Then, they apply this visual channel to several applications, including authenticated key exchange between devices and secure device configuration and pairing in smart home systems. Hayashi et al. [35] introduced WebTicket, a web account management system that employs visual tokens called tickets, consisting of 2D barcodes, to authenticate the users to a remote service. The tickets can be printed or stored on smartphones and are presented to the computer's webcam upon authentication. Pixie replaces the user action of scanning a barcode with the snapping of a photo, and may provide a faster alternative to visual token based authentication, especially when the trinket is readily accessible to the user, e.g., tattoo, piece of jewelry worn by the user, etc.

2.3 Wearable Device Authentication

To address the limited input space of wearable devices, available sensors (e.g. camera) are commonly exploited to provide alternative input techniques: Omata and Imai [52] identify the input gesture of the user by sensing the deformation of the skin under the smartwatch. Withana et al. [86] use infrared sensors to capture the gesture input of the user to interact with a wearable device. Yoon et al. [87] exploit the ambient light sensor to capture the changes in light state as a form of PIN entry for wearable devices.

Similar to Pixie, cameras integrated in wearable devices have been used to capture the input for authentication. Van Vlaenderen et al. [78] exploit the smartwatch camera to provide the device with an input (e.g. PIN) that is drawn on a canvas, then use image processing techniques to interpret the captured input. Chan et al. [13] propose to pair and unlock smartglasses with the user smartphone by exploiting the glass camera to scan a QR code that is displayed on the user's phone screen. Similarly, Khan et al. [41] use the smartglass camera to scan a QR code that is displayed on a point-of-service terminals (e.g. ATM) to connect to a cloud server for obtaining an OTP.

Wearable devices can be used as the second authentication factor, see [5] for a survey. Corner and Noble [16] use a wearable authentication token, which can communicate to a laptop over short-range wireless, to provide continuous authentication to the laptop. Lee and Lee [43] use the smartwatch to collect and send the motion patterns of the user for continuous authentication to a smartphone.

As Pixie does not require uncommon sensors or hardware, but only a camera, it is suitable for several camera equipped wearables [63, 73, 80].

2.4 Graphical Passwords

Pixie's visual nature is similar to graphical passwords, that include recall, recognition and cued-recall systems (see [7] for a survey). Recall based solutions such as DAS (Draw-A-Secret) [37] and variants [26, 30] ask the user to enter their password using a stylus, mouse or finger. For instance, De Luca et al. [22] proposed to enter the stroke based password on the front or back of a double sided touch screen device. In recognition-based systems (e.g., Passfaces [24, 53]), users create a password by selecting and memorizing a set of images (e.g., faces), which they need to recognize from among other images during the authentication process. Cued-recall systems improve password memorability by requiring users to remember and target (click on) specific locations of an image [66, 83, 84]. For instance, Schneegass et

al. [66] observed that authentication can be made resistant to fingerprint smudge attacks. Specifically, they proposed SmudgeSafe, an authentication solution that employs geometric image transformations to modify the appearance of the underlying image for each authentication attempt.

Pixie can be viewed as a recognition based graphical password system where the possible secret images are dynamically generated based on the physical world around the user. Since the user freely presents the candidate password through a photo of the physical world, captured in different light, background, and angle conditions, Pixie has to implement an accurate matching of trinkets. Trinkets can be small portions of items worn by users (e.g., shirt pattern, shoe section). Pixie accurately verifies that the candidate image contains the same trinket part as a set of previously captured reference images. This process endows Pixie with attack resilience properties: to fraudulently authenticate, an adversary needs to capture both the mobile device and the trinket, then guess the correct part of the trinket.

2.5 Text-Based Passwords

The usability of traditional text-based passwords has been well studied in literature, see e.g., [14, 48, 70, 76]. Trewin et al. [76] found that face biometrics can be entered faster than text based passwords and Table 1 shows that Pixie is also faster than text based passwords. Several limitations are associated with text passwords on memorability and usability especially when adopted in mobile platforms. For instance, Shay et al. [70] have shown through a large user study of different password-composition policies, that more than 20% of participants had problems recalling their password and 35% of the users reported that remembering a password is difficult. Their reported user entry time for text passwords ranges between 11.6-16.2s (see Table 1) in line with our evaluation (see § 6.2.4). Pixie is also perceived as more memorable than text passwords (see 6.2.5).

Melicher et al. [48] found that creating and entering passwords on mobile devices take longer than desktops and laptops. In mobile devices, text-based passwords need to be entered on spatially limited keyboards on which typing a single character may require multiple touches [64], due also to typing the wrong key. Pixie replaces typing a password with pointing the camera to the trinket and snapping a photo of it.

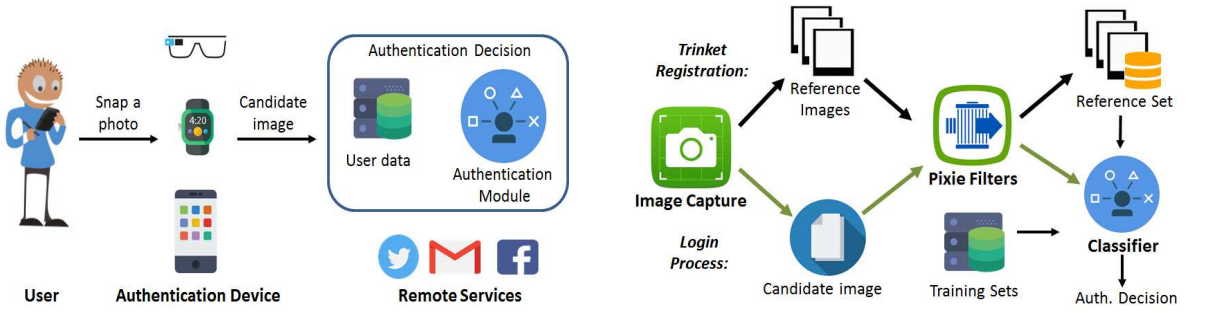
3 SYSTEM AND ADVERSARY MODEL

3.1 System Model

Figure 3(a) illustrates the system model. The user has a camera equipped device, called the *authentication device*. Authentication devices include smartphones, tablets, resource constrained devices such as smartwatches and smartglasses, and complex cyber-physical systems such as cars. The user uses the authentication device to access remote services such as e-mail, bank and social network accounts, or cyber-physical systems, e.g., home or child monitoring systems (see § 3.2 for a discussion on other related scenarios).

We assume that the user can select and easily access a physical object, the *trinket*. The user sets the authentication secret to consist of multiple photos of the trinket, taken with the device camera. We call these "reference" images, or reference set. To authenticate, the user snaps a "candidate" image of the trinket. This image needs to match the stored, reference set. Figure 3(a) illustrates an approach where the remote service stores the user's reference set and performs the image match operation. In § 7 we compare the merits and drawbacks of this approach to one where the authentication device performs these tasks.

Pixie can be used both as a standalone authentication solution and as a secondary authentication solution, e.g., complementing text based passwords.



(a) (b)

Fig. 3. (a) **System model**: the user authenticates through a camera equipped device (smartphone, smartwatch, Google Glass, car), to a remote service, e.g., e-mail, bank, social network account. The remote service stores the user credentials and performs the authentication. (b) **Pixie registration and login workflows**: to register, the user captures "reference images" of the trinket, which are filtered for quality and consistency. To authenticate, the user needs to capture a "candidate image" of the trinket that matches the reference images.

3.2 Applications

While this paper centers on a remote service authentication through a mobile device scenario, Pixie has multiple other applications such as authentication in camera equipped cyber-physical systems. For instance, cars can use Pixie to authenticate their drivers locally and to remote services [67]. Pixie can also authenticate users to remote, smart house or child monitoring systems, through their wearable devices. Further, door locks, PIN pads [65, 67] and fingerprint readers can be replaced with a camera through which users snap a photo of their trinket to authenticate.

Pixie can be used as an alternative to face based authentication when the users are reluctant to provide their biometric information (e.g. in home game systems where the user needs to authenticate to pick a profile before playing or to unlock certain functionalities). Pixie can also be used as an automatic access control checkpoint (e.g. for accessing privileged parts of a building). The users can print a visual token and use it to pass Pixie access control checkpoints.

In addition, given the large number of people who work from home [69], Pixie can provide an inexpensive 2FA alternative for organizations to authenticate employees who are connecting to the private network remotely [32]: replace the hardware tokens with user chosen Pixie trinkets.

We note however that as we discuss later, Pixie may be unsuitable in authentication scenarios that include (1) a high risk associated with external observers, (2) poor light conditions, (3) unpredictable movements, e.g., while walking or in public transportation, or (4) depending on the trinket object type, situations where the user cannot use both hands.

3.3 Adversary Model

We assume that the adversary can physically capture the mobile device of the victim. We also assume that the adversary can use image datasets that he captures and collects (see § 5.1) to launch brute force **pictionary attacks** against Pixie (see § 5.3.1).

Similar to PIN based authentication to an ATM, Pixie users need to make sure that onlookers are far away and cannot see the trinket and its angle. We assume thus an adversary with *incomplete surveillance* [28], who cannot observe or record the trinket details. However, we consider a **shoulder surfing** attack flavor where the adversary sees or guesses the user's trinket object type. The adversary can then use datasets of images of similar objects to attack Pixie (see § 5.3.2).

Further, we also consider an adversary that attempts to launch a **master image attack**, i.e., identify images that contain diverse features and match many trinkets. Example master images include "clutter" images, with an array of shapes, colors and shadows (see § 5.3.3).

4 PIXIE

4.1 Pixie Requirements

Pixie is a two factor authentication solution as it requires both a possession factor and a knowledge factor to authenticate the user. The possession factor is the trinket. The knowledge factors are the trinket, and its angle and section used to authenticate. In addition to being resilient against attacks (see § 3.3), Pixie needs to satisfy the following requirements:

- **Trinket image quality**. Pixie needs to ensure the quality of trinkets and images. Early pilot studies showed that not all the trinkets that the users chose, or the photos that they took, were suitable for authentication.
- **Trinket match**. Pixie needs to match images of the same trinket, even when captured with a different background, lighting, or from a slightly different distance or angle.
- **Discoverability**. New users should easily discover the functionality of Pixie.
- **Deployability**. Pixie should be easy to integrate into existing systems.

Figure 3(b) depicts the modular approach we use for Pixie to address these goals. The image capture module seeks to address part of the first requirement, by facilitating the capture of quality trinket images. The authentication module tackles the second requirement through the use of trained classifiers to match trinket images. To simultaneously address the first and third requirements, i.e., to ensure the discoverability of Pixie while guiding new users through the capture of high quality photos and the choice of visually complex objects as the secret, the filter module detects and eliminates low quality images and invalid reference sets. We now detail each module.

4.2 Image Capture & Feedback

We performed pilot studies to identify early problems with the Pixie user interface. For instance, during the pilot studies, some users captured trinket photos whose background provided more features than the trinkets. This revealed that the trinket needs to be the main object in captured images. To simultaneously satisfy this requirement, and the trinket quality requirement above, we design Pixie to guide the user to take larger photos of trinkets. We achieve this by overlaying a circle on the camera image: the user needs to fit the trinket impression inside the circle (see Figures 1(a) and 1(b)). Since Pixie does not allow zooming in, the user needs to bring the camera closer to the trinket, hence take a larger photo. Pixie crops the image, and keeps only the largest rectangle parallel to the sides of the device that fits the circle.

In addition, we observed that the quality of trinket images captured by the users could be low (e.g. blurry or dark), or the users may take inconsistent trinket images in the registration phase. To ensure the quality of trinket images and the consistency of reference images, we identified common problems that occur during the image capture process (e.g., insufficient light, trinket with plain texture). Then, we mapped prefilter rejection decisions provided by Pixie's image filter (see § 4.4) into informative error messages (see Figure 1(c)). Furthermore, to facilitate the discoverability of Pixie, we designed and included a step by step in-app instruction guide on how to use Pixie. Table 2 summarizes the design improvements we made to the Pixie UI.

Table 2. Summary of user interface improvements identified during pilot studies.

Requirement	Pixie Feature
Increase the size of trinket & Reduce the background area in trinket images	1. Disable camera zoom 2. Overlay a circle as the target area on the camera view
Ensure the quality of reference and candidate images Ensure consistency of reference images	1. Design prefilters for checking the quality of images 2. Translate the prefilter criteria into actionable feedback to the users
Improve the discoverability of Pixie	1. Show number of remaining images to take in registration screen 2. Show camera capture icon for login page 3. Add step by step in-app instruction on how to use Pixie

Table 3. Pixie notations and algorithm acronyms.

Symbol	Description
$\overline{R}$	The set of reference images
R	Any of the reference images in the reference set ($\overline{R}$)
C	The candidate image
$T(\overline{R})$	Template image of reference set ($\overline{R}$)
$NNSim(C, \overline{R})$	Nearest neighbor similarity of C to $\overline{R}$
$FNSim(C, \overline{R})$	Furthest neighbor similarity of C to $\overline{R}$
$AvgRefNN(\overline{R})$	Avg. nearest neighbor similarity of each reference image
$AvgRefFN(\overline{R})$	Avg. furthest neighbor similarity of each reference image
$AvgReflTempl(\overline{R})$	Avg. similarity of reference images to template image
KP-CNT	Number of keypoints in an image
DTC-KP	Avg. distance of keypoints to their centroid in an image
White-CNT	Number of detected edge (white) pixels of an image
DTC-White	Avg. distance of edge (white) pixels to their centroid
$MinCrossSim(\overline{R})$	Min. similarity among all the pairs of images in $\overline{R}$
$MaxCrossSim(\overline{R})$	Max. similarity among all the pairs of images in $\overline{R}$
$AvgCrossSim(\overline{R})$	Avg. similarity among all the pairs of images in $\overline{R}$
ORB [60]	ORB keypoint extraction algorithm
SURF [3]	Speeded Up Robust Features keypoint extraction algorithm
RANSAC [29]	Random Sample Consensus algorithm for fitting the model to data
FLANN [50]	Fast Approximate Nearest Neighbor Search

4.3 The Authentication Module

The authentication module is responsible for addressing Pixie's second requirement (see § 4.1), of matching the candidate image against the reference images. Pixie extracts robust keypoints from these images, identifies a suite of features from the keypoint match process, then uses them to train a classifier that decides if the candidate image matches the reference set. We now detail this process. Table 3 summarizes the most important Pixie features notations. Let C denote the candidate image, $\overline{R}$ be the set of reference images, and R be any of the reference images (see § 3.1).

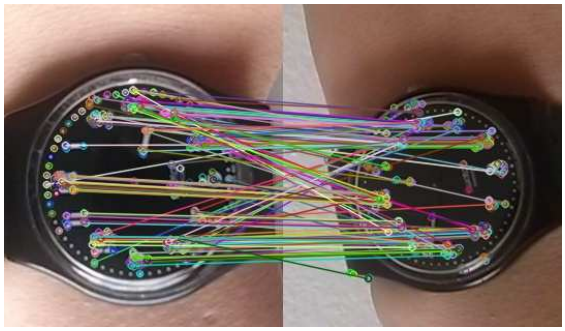


Fig. 4. Example ORB keypoint matches between two images of the same trinket, taken in different conditions. **Each line represents a match**: it connects matching keypoints (shown as small colored circles) from each image.

Keypoint matching. We use SURF (Speeded Up Robust Features) [3] and ORB [60] algorithms, to extract scale and rotation invariant image keypoints from the candidate and reference images, e.g., shown as small colored circles on images in Figure 4 and 2. We also extract the descriptors of the keypoints, which represent their characteristics. To determine if a candidate image C and a reference image R contain the same trinket, we compute a 1-to-1 matching between their keypoint descriptors (e.g., shown as lines in Figure 4). We use brute-force matching for ORB keypoints, where each keypoint of the candidate image is matched with the closest keypoint (in terms of Hamming distance) of the reference image. For SURF keypoints, we use the FLANN-based matcher [50].

An exhaustive matching of each keypoint in the candidate image to a keypoint in the reference image will produce low quality, *outlier* matches. We experimented with several existing filters, including threshold, cross checking and RANSAC [29], to identify and remove outlier matches. The RANSAC based filter performed the best, hence we use it implicitly in the following.

Image similarities. Given two images C and R, we define their similarity $Sim(C, R)$ to be the ratio between the number of keypoint matches of C and R, after the above filter and outlier detection steps, and the number of keypoints in C. Given C and the set $\overline{R}$, we define the nearest neighbor similarity of C to $\overline{R}$ as $NNSim(C, \overline{R}) = \max\{Sim(C, R)|\forall R \in \overline{R}\}$, and the farthest neighbor similarity, $FNSim(C, \overline{R}) = \min\{Sim(C, R)|\forall R \in \overline{R}\}$.

Given a reference set $\overline{R}$, we define the average nearest neighbor similarity value of each reference image, to the other reference images in $\overline{R}$: $AvgRefNN(\overline{R}) = \frac{\Sigma_{R\in\overline{R}} NNSim(R,\overline{R}-R)}{|\overline{R}|}$. Similarly, we define the average farthest neighbor similarity value of each reference image to the other images in $\overline{R}$: $AvgRefFN(\overline{R}) = \frac{\Sigma_{R\in\overline{R}} FNSim(R,\overline{R}-R)}{|\overline{R}|}$.

Template image. Given a reference set $\overline{R}$, we define its *template image*, $T(\overline{R})$, as the reference image R whose value $\sum_{r\in\overline{R}-R} Sim(r, R)$ is the maximum among all reference images in $\overline{R}$. Intuitively, $T(\overline{R})$ is the reference image "closest to the center" of the reference set. We define $AvgRefTempl(\overline{R})$ as the average similarity of images in $\overline{R}$ to $T(\overline{R})$.

Pixie matching features. We use the above concepts to extract the following features (see Table 4, top section). We use these features to train a supervised learning algorithm.

• *Keypoint counts*. The keypoint count of C and $T(\overline{R})$.

• *Match based features*. The number of keypoints in C and $T(\overline{R})$ that match, before the RANSAC filter. The min, max, mean and SD of the distance, size, response and angles between the matched keypoints in C and $T(\overline{R})$, after RANSAC.

Table 4. Summary of (top) Pixie features and (bottom) Pixie filter features.

Solution	Features	Details
Pixie	Keypoint stats.	Statistics of ORB/SURF keypoints
	Keypoint nearest neighbors	Keypoint match stats.
	Perspective transformation	RANSAC optimal map of match keypoints
Pixie filters	Keypoints	Count and spread of keypoints
	Edge pixels	Count and spread of edge pixels
	Reference quality	Reference image similarities stats.

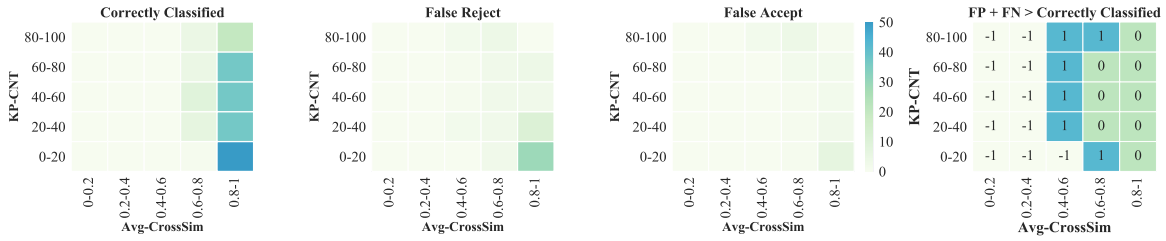
• *Quality of the reference set.* $AvgRefNN(\overline{R})$, $AvgRefFN(\overline{R})$ and $AvgReflempl(\overline{R})$.
• *Similarity to template.* The similarity of C to $T(\overline{R})$, normalized by the average similarity of the images in $\overline{R}$ to $T(\overline{R})$, i.e., $\frac{Sim(C,Templ(\overline{R}))}{AvgRefTempl(\overline{R})}$.
• *Similarity to reference set.* We define $minSim(C,\overline{R}) = \frac{min\{Sim(C,R)|\forall R \in \overline{R}\}}{AvgRefFN(\overline{R})}$: the ratio of the similarity between C and "farthest" reference image, and the average least similarity between reference images. Similarly,
$maxSim(C,\overline{R}) = \frac{max\{Sim(C,R)|\forall R \in \overline{R}\}}{AvgRefNN(\overline{R})}$.
• *Homography*: Output of homography between C and $T(\overline{R})$: the perspective transformation between the planes of the two images (3 features).

4.4 Pixie Filters

Early pilot studies revealed that Pixie users can capture low quality images. Such images, either reference or candidate, hinder the ability of the authentication module to discern candidate images, increasing the FRR of Pixie. Furthermore, they impose gratuitous network latency in the remote authentication scenario (see § 3.1).

Several conditions may prevent taking high quality images Figure 2(d)-(f) shows example outcomes of such conditions, including (i) improper lighting, (ii) unsteady hand and (iii) choice of trinkets with constant texture. We also observed that some pilot study participants, during the reference set registration process, took photos containing different trinkets, or different areas of the same trinket. To address these issues, we introduce a set of filters (see Figure 3(b)) that reject problematic images captured by the user. We propose the *two rules of filtering*, that set out the operation space for Pixie image filters:
• **Filter Rule #1**: Pixie may not willfully fail by operating on images on which it predicts it will fail.
• **Filter Rule #2**: Pixie may not operate in a space where it has not been trained.
In the following, we detail these rules and describe the resulting filters.

4.4.1 Filter Rule #1: CBFilter and RBFilter. We introduce CBFilter and RBFilter, filters that identify reference and candidate images on which they predict Pixie will fail. The filters leverage the following features, (see Table 4(bottom section) for a summary).
Filter features. First, we define KP-CNT as the keypoint count of an image. The intuition for using this feature is that an image with a low KP-CNT (e.g., Figure 2(f) with only 5 keypoints) is likely to negatively impact the accuracy of Pixie's matching process. A second feature is based on the center, or *centroid* of the keypoints extracted from an image: let DTC-KP (distance to center of keypoints) denote the average distance between the keypoints of the image and their centroid. DTC-KP measures the spread of the keypoints across the image. The intuition is that a high DTC-KP may indicate that some



(a) (b) (c) (d)

Fig. 5. Example 2D histograms of KP-CNT of template image vs. $AvgCrossSim(\overline{R})$. (a) Correctly classified instances. (b) False reject instances. (c) False accept instances. The legend in (a)-(c) shows the color code used for the number of authentication instances. (d) Aggregated 2D histogram. The darker regions with 1 in the center have a greater proportion of misclassified than correctly classified instances. The regions with -1 in the center correspond to value ranges on which we have no template images. Conclusion: filter out reference sets with KP-CNT < 20 and $AvgCrossSim(\overline{R}) < 0.6$.

keypoints do not belong to the trinket but to the background. Third, to detect blurry images, we use the Canny edge detector [9] to identify edge pixels that delimit objects in the image. Let White-CNT denote the number of detected edge ("white") pixels of an image. White-CNT is an indicator of the clarity of the image: a low White-CNT denotes a blurred image, with few trinket edges. We also introduce DTC-White (distance to center of white pixels), the average distance of the white pixels to their centroid. DTC-White denotes the spread of the edge pixels, i.e., the size of the trinket. Finally, to detect inconsistent reference images, we define $MinCrossSim(\overline{R})$, $MaxCrossSim(\overline{R})$ and $AvgCrossSim(\overline{R})$, to be the minimum, maximum and average similarity (see § 4.3) among all the pairs of images in $\overline{R}$. Small cross similarity values indicate reference images of non-identical trinkets.

CBFilter: Classifier Based Filter. Given the reference set $\overline{R}$ and its template image $T(\overline{R})$ (see § 4.3), CBFilter uses a suite of features to train a supervised learning algorithm and determine if $\overline{R}$ is suitable to participate in the authentication process. The features include KP-CNT, DTC-KP, White-CNT, DTC-White of $T(\overline{R})$, the average, minimum and maximum of KP-CNT, DTC-KP, White-CNT, DTC-White over all the images in $\overline{R}$, and $MinCrossSim(\overline{R})$, $MaxCrossSim(\overline{R})$ and $AvgCrossSim(\overline{R})$.

RBFilter: Rule Based Filter. Pilot studies demonstrated the need to give relevant feedback to users as early as possible: early pilot study participants expressed frustration when they discovered that the photos they took were not suitable at the end of the registration, or worse, during the authentication process. The output of CBFilter cannot however be used to provide meaningful feedback.

To address this limitation, we identified common problems that occur during the image capture process, e.g., improper light, trinket with plain texture or not identical reference images. We then developed a set of rules for these filter features, that (i) predict if an image or image set will not perform well during authentication, and (ii) that can be transposed to one of the problems identified. For instance, we found that a small KP-CNT is associated with insufficient light, blur, or trinkets with a plain texture, while a small $AvgCrossSim$ value can indicate reference images containing non-identical trinkets. Figure 1(c) illustrates the feedback provided when the user captures a low quality trinket (top) or inconsistent reference images (bottom).

To identify such rules, we run Pixie on the Pixie dataset, a dataset of reference set and candidate image pairs that are captured in different conditions (see § 5.1 for more details). Specifically, we investigate reference sets and candidate images that contributed to misclassified instances as follows. For each pair of the above filter features, we plot the 2D histogram of instances that were correctly classified, and that contributed to false accepts (FA) and false rejects (FR). Figures 5(a)-(c) illustrates this process for the KP-CNT of template images $T(\overline{R})$ vs. $AvgCrossSim(\overline{R})$ pair of features. Then, we aggregate the results

Table 5. RBFilter and UBounds filter rules for reference and candidate images, and their real world interpretation. RBFilter (top 2 sections) filters images on which it predicts Pixie will fail. UBounds (bottom section) filters images outside the space seen by Pixie during training.

Image type	Filter Rule	Interpretation
Reference	KP-CNT < 20	Low quality image or plain trinket
Reference	DTC-KP < 30	Low quality image or plain trinket
Reference	AvgCrossSim < 0.6	Non-identical trinkets in reference set
Candidate	KP-CNT < 20	Low quality image or plain trinket
Candidate	DTC-KP $> 44,600$	Out of bounds image
Candidate	White-CNT $> 22,400$	Out of bounds image
Candidate	DTC-White > 160	Out of bounds image

for the three 2D histograms, see Figure 5(d), by calculating the contribution of each type of classification result (i.e., FA, FR, True Accept (TA) and True Reject (TR)) in a cell of the 2D histogram. The dark regions have a larger proportion of misclassified than correctly classified instances. This enables us to identify "problem" regions, where the contribution of misclassified instances (FA and FR) is larger than that of correctly classified instances (TA and TR). We then define rules, i.e., threshold values, that avoid clusters of problem regions. For instance, based on the bottom area of Figure 5(d), we reject reference sets whose template has KP-CNT < 20. Similarly, we reject reference sets with $AvgCrossSim(\overline{R}) < 0.6$, as we have none with $AvgCrossSim(\overline{R}) < 0.4$ (cells with -1), and those in $[0.4, 0.6]$ are frequently misclassified.

Through a similar process, we have identified several other filtering rules for reference sets and candidate images, and their real world interpretation, see Table 5 (top 2 sections). RBFilter uses these rules to reject low quality reference and candidate images, and extend Pixie with informative error messages that guide users to improve the quality of captured images.

4.4.2 Filter Rule #2: UBounds. We train Pixie on a dataset of images that do not cover the entire value space of the filter features. Pixie cannot make informed decisions on candidate images whose features take values in sub-areas not seen during training. We have identified several such sub-areas for the Pixie dataset. The UBounds filter consists of the "universe boundary" rules listed in Table 5 (bottom section), that define these sub-areas.

5 EVALUATION

We have implemented Pixie using Android 3.2, OpenCV 2.4.10 and Weka [82]. In the following we first evaluate the performance of the Pixie features, and parameters, under several supervised learning algorithms. We then evaluate the performance of Pixie's optimal configuration under the attacks introduced in § 3.3. We report the performance of Pixie through its False Accept Rate (FAR), False Reject Rate (FRR), Equal Error Rate (EER), the rate at which $FAR = FRR$, F-measure, and Failure to Enroll (FTE). For our experiments, we have used a Mac OS X (2.9 GHz Intel Core i7 CPU, and 8GB DDR3 RAM) and a Nexus 4 smartphone (Quad-core 1.5 GHz Krait, and 2GB RAM; 8MP camera sensor, f/2.4 aperture).

5.1 Data

We have collected and generated the following datasets:

Nexus image dataset. We used a Nexus 4 device to capture $1,400$ photos of 350 unique trinkets, belonging to 33 object categories. We selected only objects that can be easily carried by users and are thus ideal candidates for image-based trinkets, e.g., watches, shoes, jewelry, shirt patterns, credit cards

Table 6. ORB vs. SURF based Pixie (MLP classifier, no filter) performance, on the Pixie dataset. SURF has lower FAR and FRR compared to ORB.

Keypoint Detector	FAR(%)	FRR(%)	F-measure(%)	EER(%)
ORB	0.10	9.83	93.08	4.87
SURF	0.07	4.80	96.40	2.80

and logos. We have captured 4 images for each trinket, that differ in background and lighting conditions, i.e., either indoors using artificial light or outdoors in daylight conditions.

Pixie dataset. To evaluate Pixie, we generate authentication instances that consist of one candidate image and 3 reference images. To prevent "tainting", we need to ensure that instances used for testing do not contain reference images that have appeared in a training instance. For this, we use the $1,400$ images of the 350 trinkets, to generate 10 Pixie subsets, each containing 10 folds, as follows. To generate one of the 10 folds of one of the 10 subsets, we first randomly split the 350 trinkets into 10 sets of 35 trinkets each. For each trinket in a set, we randomly select one of its 4 images as candidate; the remaining 3 images are reference images. The trinket then contributes to the fold by one genuine instance (its candidate + its 3 reference images) and 34 "fraud" instances. Each fraud instance combines the trinket's candidate image with the 3 reference images of one of the other 34 trinkets in the subset. Thus, each fold consists of 35 authentic and $1190 = 35 \times 34$ fraud instances. Then, one of the 10 Pixie subsets contains $12,250$ authentication instances. Thus, the Pixie dataset has a total of $122,500$ authentication instances.

Google Image dataset. We used Google's image search site to retrieve at least 200 images from each of the 33 object categories of the Nexus image dataset, for a total of $7,853$ images. This dataset forms the basis of a shoulder surfing attack (see § 5.3).

ALOI dataset. We use the illumination subset of the Amsterdam Library of Object Images (ALOI) [31] dataset, that contains 24 different images for 1000 small objects (i.e., natural trinket choices) captured under various illumination conditions. We cropped these images to the size of the Nexus images (270×312 pixels), while keeping their object centered.

Caltech101 dataset. Caltech101 [27] is a collection of $9,145$ images from 101 object categories.

5.2 Parameter Choice for Pixie

We first identify the parameters where Pixie performs best.

ORB vs. SURF. We compare the performance of Pixie when using two popular keypoint extraction algorithms, ORB [60] and SURF [3]. We use Multilayer Perceptron (MLP) for the Pixie classifier, and no filter. We perform the evaluation through 10-fold cross validation on each of the 10 subsets of the Pixie dataset (see § 5.1). Table 6 reports the performance of ORB and SURF: SURF has lower FAR and FRR, leading to an EER that is smaller by 2% than that of ORB.

However, ORB is faster than SURF: in an experiment on 100 Nexus images, ORB took an average 0.15s to extract keypoints on the Nexus 4, while SURF took 2.5s on the Mac and almost 5s on the Nexus 4. ORB's keypoint match is also faster: in an experiment over $10,000$ image pairs, on the Nexus 4, SURF's keypoint match took an average of 2.72s, while ORB took 0.66s.

Given the trade-off between speed and accuracy, SURF is more suitable when the image processing and matching tasks can be performed on a server. The faster ORB should be preferred in the mobile authentication scenario, when these tasks have to be performed by a mobile device. In the following experiments, we set Pixie's keypoint extraction algorithm to be ORB.

Classifier Choice. We use the Pixie dataset to identify the best performing classifier for Pixie's authentication module. Table 7 shows the results: Random Forest (RF) and MLP outperform Support Vector

Table 7. Classifier performance on Pixie dataset using ORB keypoint extractor and no filter. **Random Forest and MLP achieve the lowest EER**, thus we only use them in the following.

Pixie Classifier	FAR(%)	FRR(%)	F-measure(%)	EER(%)
MLP	0.10	9.83	93.08	4.87
RF	0.02	10.74	93.90	3.82
SVM	0.00	12.57	93.04	10.74
Decision Tree (C4.5)	0.17	11.54	91.01	7.66

Table 8. Performance of Pixie MLP classifier with RBFilter on the Pixie dataset. The disjunction of all the RBFilters on the reference images reduced the FAR and FRR by more than 40%.

Images	Filtering Rule	FAR(%)	FRR(%)	F-measure(%)
Reference	KP-CNT <20	0.09	6.60	95.06
Reference	DTC-KP <30	0.10	9.12	93.46
Reference	AvgCrossSim <0.6	0.07	8.10	94.53
Reference	All 3 Filters	0.06	4.46	96.75
Ref. & cand.	All RBFilter Rules	0.04	5.25	96.58

Table 9. Pixie + CBFilter performance, for various combinations of supervised learning algorithms. **CBFilter is effective: when using RF, it reduces the EER of Pixie (with MLP) to 1.87%.**

Pixie	CBFilter	FAR(%)	FRR(%)	F-measure(%)	EER(%)
MLP	MLP	0.07	6.34	95.54	2.97
MLP	RF	0.06	4.70	96.52	1.87
MLP	C4.5	0.02	7.19	95.92	2.35
MLP	SVM	0.10	9.83	93.08	4.87
RF	MLP	0.02	7.64	95.63	2.72
RF	RF	0.01	5.74	96.77	1.96
RF	C4.5	0.02	7.19	95.92	2.35
RF	SVM	0.02	10.74	93.90	3.82

Machine (SVM) and Decision Tree (DT) through lower FAR and FRR. In the following, we use only RF and MLP as Pixie's classifiers.

Pixie with RBFilter. We evaluate the effects of the RBFilter rules of Table 5 on the performance of Pixie. For this, in each of the 100 classification experiments (10 folds cross validation over each of the 10 subsets of the Pixie dataset), we remove from the Pixie test fold all the authentication instances that satisfy the rules. We then run Pixie (with MLP) on this filtered dataset.

Table 8 shows that all the rules are effective: each increases Pixie's F-measure. The disjunction of all the reference set filter rules is the most effective, for an F-measure of 96.75% (3.8% improvement from the unfiltered 93.08% of Table 6). The 3 reference set filter rules remove an average of 6.68 reference sets from a testing fold. When also using the candidate image filter, that removes an average of 82.23 authentication instances per testing fold, Pixie's F-measure drops to 96.58%. This is because we count the "valid" instances removed by the candidate filter as part of FRR, even though they are likely of low quality and can mislead Pixie.

Table 10. Filters effects on Pixie performance. The combination of RBFilter and CBFilter (RF) has the best performance.

Algo	FAR(%)	FRR(%)	F-measure(%)
Pixie	0.10	9.83	93.08
Pixie & RBFilter	0.04	5.25	96.58
Pixie & CBFilter	0.06	4.70	96.52
Pixie & RBFilter & CBFilter	0.02	4.25	97.52

Pixie with CBFilter. To provide a large training set for CBFilter, we first build a Reference Set Bank (RSB), that contains all the reference sets that appear in the 10 subsets of the Pixie dataset. For each such reference set, the RSB also stores its "class", according to the outcome of Pixie: if the reference set has been part of any authentication instance (in the Pixie dataset) that was incorrectly classified by Pixie (i.e., either as FR or FA), its class is 1, otherwise it is 0.

We use the RSB set for the following evaluation process, performed separately for each subset of the Pixie dataset. Each of the subset's 10 folds, is used once for testing. Given one such fold, e.g., F_1, we extract its reference sets. We train CBFilter on all the reference sets of RSB, that are different from the reference sets of fold F_1, then test CBFilter on the reference sets of F_1. We filter from F_1 all the reference sets that are labeled as 1 by CBFilter. Finally, we train Pixie on the 9 other folds ($F_2..F_{10}$) and test it on the filtered F_1. We repeat this process 100 times (for the 10 folds of each of the 10 subsets of the Pixie dataset).

Table 9 compares the performance of various classifiers for both Pixie and CBFilter. It shows that CBFilter is effective: when using Random Forest, it reduces the EER of Pixie to 1.87% (from 4.87%), and removes 3.45 reference sets on average from a testing fold.

Pixie, RBFilter and CBFilter. When used in combination with RBFilter, CBFilter removes an additional 0.9 reference sets on average from a testing fold. RBFilter's candidate rule also removes 79.59 instances. Table 10 compares the performance of the combined Pixie, RBFilter and CBFilter against the performance of the unfiltered Pixie, as well as Pixie's combination with only one of the filters. When used together, the filters reduce the FAR of the basic Pixie by 80% and its FRR by 56%.

Comparison to other authentication methods. The performance of Pixie (EER=1.87) compares favorably with the performance of other biometric based authentication solutions. For instance, Meng et al. [49] report EERs of 2-4% and 2-6% for authentication solutions based on face and fingerprint. Samangouei et al. [62] report EERs of 13-30% for attribute based face authentication, and Taigman et al. [75] report an EER of 8.6% for face recognition using features extracted by deep neural networks. The gaze-challenge authentication solution of Sluganovic et al. [72] has an EER of 6.3%, while Zhao et al. [88] report EERs between 4.1-9.6% for touch gesture based authentication.

5.3 Pixie Under Attack

We investigate the performance of Pixie, trained on one of the 10 Pixie subsets, under the attacks of § 3.3. We use the previously identified parameters: the ORB keypoint extractor, MLP for the Pixie classifier, RF for the CBFilter classifier, and all the rules for RBFilter. Using UBounds filter we obtain a conservative performance of Pixie: with UBounds, Pixie would easily reject out of bounds images, artificially boosting its accuracy.

Attack datasets. We use the Nexus dataset (§ 5.1) to build 3 *authentication attack datasets* based on the ALOI, Google Image and Caltech101 sets. We use the Google Image based attack dataset for

a shoulder surfing attack, and, along with the ALOI and Caltech101 datasets, to evaluate brute force pictionary attacks.

We generate the authentication attack instances for each attack dataset, and group them into 10 folds, as follows. We randomly split the 350 unique trinkets of the Nexus dataset into 10 subsets of 35 trinkets each. For each trinket in a subset, we randomly select 3 out of its 4 images, to form a reference set. We then combine this set with each of the images from ALOI, Google Image, and Caltech101 datasets, respectively. We repeat this process for all the 35 reference sets in a fold. Thus, in the ALOI attack dataset, a fold contains 840K = 35 × 24K attack instances, for a total of 8.4M ALOI based attack instances. Similarly, the Google Image attack dataset contains 2.7M+ attack instances, while the Caltech101 attack dataset contains 3.2M+ instances.

Table 11. Performance of Pixie (with RBFilter and CBFilter) on the ALOI, Caltech101 and Google attack datasets: **On more than 14M attack authentication samples, the FRR of Pixie is less than 0.09%.**

Attack Dataset	FAR(%)
Google	0.054
ALOI	0.087
Caltech101	0.042

5.3.1 Pictionary Attack. Under the Google Image attack dataset, Pixie achieved a FAR of 0.054%, see Table 11. 216 of the 350 trinkets were not broken. However, we counted each such trinket as success at 7,853 trials. Then, the average number of Google dataset based "trials until success", over the 350 trinkets is 5,766.12. For the ALOI based attack, when using both RBFilter and CBFilter, Pixie achieved a FAR of 0.087%. Under the Caltech101 attack, Pixie's FAR is 0.042%. The higher FAR of the ALOI pictionary attack dataset may be due to the similarity of its images of small objects to images in the Pixie datase. Pixie filters about 10 reference sets from each attack dataset. In addition, it filters a small number of candidate images (82 and 5) from the Google and Caltech101 datasets, but 1,449 candidate images from the ALOI dataset.

5.3.2 Restricted Shoulder Surfing Attack. We use the Pixie and Google Image datasets to evaluate the "guessing entropy" [19] of the restricted shoulder surfing attack. The attack proceeds as follows: for each reference set of a Pixie dataset trinket, we re-order the Google dataset images to start the brute force attack with images of the same type as the trinket. We then use each image in the re-ordered Google dataset as candidate, and count the number of trials before a match (false accept) occurs. Thus, this experiment evaluates the scenario where the adversary exploits his knowledge of the trinket type.

As in the pictionary attack above, we counted each of the 216 unbreakable trinkets as "success" at 7,853 (the size of the attack dataset) trials. Then, the average number of "trials until success", over the 350 Pixie dataset trinkets was 5,639.53. This result is similar to the above pictionary attack: in fact, an unpaired t-test did not find a statistically significant difference in the number of trials to break a reference set between the two scenarios ($p - value = 0.44$, for $\alpha = 0.05$). Thus, in our experiments, knowledge of the trinket type does not provide the adversary with a significant guessing advantage.

5.3.3 The Master Image Attack and Defense. We identified 788 master images in the ALOI dataset, 75 in the Caltech Image dataset, and 127 in the Google dataset. Master images match multiple Pixie reference sets. Upon manual inspection, we observed that master images are not of the same type of trinket as the reference set that they match. Instead, they contain an array of shapes, shadows, letters and colors, that translate into a diverse sets of keypoints, see Figure 6 for examples. Less than half of the

Fig. 6. Example master images for Pixie: each of these images matches multiple reference sets of the Pixie dataset. Master images tend to have a rich combination of shapes, shadows, colors and letters.

master images in the ALOI (224), Caltech101 (34) and Google (30) datasets match at least 5 reference sets. 1 master image in the Caltech101 dataset matches 51 reference sets.

Defense. The shape formed by the matched keypoints in a master image is likely to be inconsistent with that of the "victim" reference set. We leverage this observation to introduce several new features: the distance to the centroid of the matched keypoints (DTC-MKP) in the candidate and template images, and the min, max and mean of the DTC-MKP over all pairs of candidate and reference images. We train the Pixie classifier using this enhanced feature set, and test it on the ALOI attack dataset. The enhanced Pixie reduces the number of effective ALOI master images (matching at least 5 reference sets) by 60%, i.e., from 224 to 88.

To evaluate the effect of the new features on the FRR, we run Pixie with both RBFilters and CBFilters on the 10 Pixie data subsets (see § 5.1) in a 10-fold cross validation experiment similar to that of § 5.2. We observed that when new features are included in the classification task, the FRR of Pixie decreases slightly from 4.25% (last row in Table 10) to 4.01%, while its FAR remained unchanged (0.02%). We conclude that the newly added features do not increase Pixie's FRR.

6 USER STUDY

We have used a lab study to evaluate the usability of Pixie's trinket based authentication and compared it against text-based passwords. In this section, we describe the methodology and results.

6.1 Design and Procedure

We performed a within-subjects study, where all the participants were exposed to every conditions considered. Specifically, the conditions were to authenticate from a smartphone to the Florida International University Portal Website (MyFIU), using (i) their username and text-based password and (ii) their Pixie trinket. MyFIU is a site that provides students with information about class schedules and administrative functionality.

We have recruited participants from the university campus over e-mail lists, bulletin boards and personal communications. All the participants were students enrolled at the university. The reason for selecting students for the study was to ensure a consistent and familiar login procedure to remote services. The participants in our study achieved text password authentication times on par with previously reported results (see § 6.2.4). Considering the ubiquity of mobile devices, we believe that the participants had no unfair advantage when compared to other social groups of similar age, with respect to their ability to perform the basic action of snapping a picture with a smartphone.

In the following, we first present some demographic information about the (n=42) participants in this study, then describe the procedure we used to perform the user study.

Table 12. Participant demographics. We chose only students in order to have a consistent experience for remote authentication (on the university portal website, MyFIU).

Demographic	Number	Proportion (%)
Gender		
Female	11	26
Male	31	74
Age		
Min	18	
Max	50	
Median	28	
Android	20	48
iPhone	21	50
Windows phone	1	2
Undergraduate	16	38
Graduate	26	62
CS/IT	38	90
Other majors	4	10
Use phone to login to remote services?	41	98

Demographics. We have recruited 42 participants for our lab study. Table 12 shows the demographics of the participants, obtained through the study questionnaires. In addition, 41 (98%) participants said they use their phones to login to their online accounts.

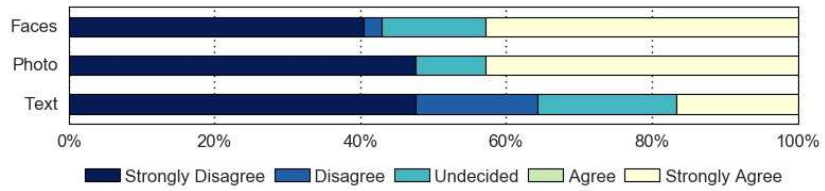
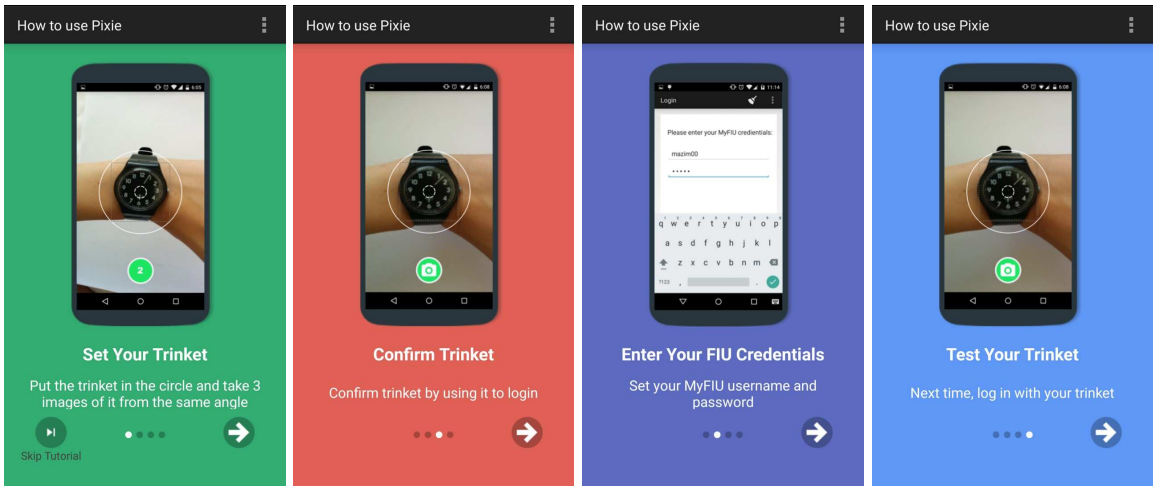


Fig. 7. Pre-study level of agreement of the participants with ease of remembering faces, photos and text. 42% of the participants strongly agree to their ease of remembering photos and faces vs. only 16% who agreed it is easy for them to remember text.

Prior to the study, we also asked the participants to express their level of agreement using a 5-point Likert scale (from 1-strongly disagree to 5-strongly agree) with how easy it is for them to remember text, photos and faces. Figure 7 shows the summary of the participants responses. More than 42% of the participants strongly agreed that it is easy for them to remember faces and photos. However, only 16% of the participants strongly agreed it is easy for them to remember text. While 64.29% of the participants said it is not easy for them to remember text, a lower 47.62% and 42.86% of the participants said it is not easy for them to remember photos and faces respectively. A pairwise non-parametric Wilcoxon-Mann-Whitney test revealed no significant difference between the perceived memorability for different items. Based on this analysis and given the picture superiority effect [51], we posit that memorizing trinkets and their secret angles could be perceived to be as memorable as faces and text. We compare the perceived memorability of trinkets and text passwords in § 6.2.3.



(a) (b) (c) (d)

Fig. 8. Pixie in-app instructions (best viewed in color) showing how to (a) setup a trinket, (b) confirm the trinket, (c) enter credentials for the MyFIU account the first time the app is used, and (d) login using the trinket.

The study procedure. We have conducted the study in an indoor lab using the existing artificial lighting. For the authentication device, we have used an HTC One M7 smartphone (1.7GHz CPU, 2.1 MP camera with f/2.0 aperture, 4.7 inch display with 1920 × 1080 resolution, and 137.4 × 68.2 × 9.33mm overall size).

The study consisted of 3 sessions, taking place on day 1, day 3 and day 8 of the experiment. From the total of 42 participants, 31 participants returned for and completed session 2 (7 female). Due to scheduling constraints, 3 participants returned for session 2 on day 4 or 5. 21 participants returned for and completed session 3 (4 female). The lab sessions proceeded as follows.

In the first session, we briefed participants about the purpose of the study: to explore the usability and the user interface design of a mobile device application. Then we asked them to use Pixie to login to their MyFIU account, using their credentials (username and password). Pixie associates the text credentials with the trinket's reference images. During subsequent login sessions, the users only needed to correctly capture the image of their trinket in order to access their account. Our goal was to let the participants experience Pixie for authentication, thus we did not ask them to enter their text password in subsequent sessions. As a result, the comparison of Pixie with text passwords is based only on the data collected in session 1.

Subsequently, the first session consisted of 3 steps. In the *discoverability step*, we gave no verbal instructions to participants. Instead, we asked each participant to try to figure out how to use Pixie, given only the in-app instructions, that show a watch as a trinket example. Figure 8(a-d) shows snapshots of Pixie app instructions for setting up a trinket, verifying the trinket, setting up the MyFIU account when the app is used for the first time and login using trinket.

In the *training step*, we explained Pixie's purpose and walked the participant through the process of setting and testing a trinket using a gum pack. However, we neither justified why we chose this trinket, nor specified what other objects can be used as trinkets. We then asked the participants to set a trinket for the rest of the study.

In the third, *repeatability step* we asked the participant to repeat the login part of the process. To avoid input based on muscle memory, we distracted the participant's attention between the second and third step by playing a game for 5 minutes.

In session 2 and 3, the participants were asked to login to their MyFIU account with the trinket they chose in session 1. At the end of each session, the participants filled out questionnaires that use Likert scales (ranging from 1-strongly disagree to 5-strongly agree). The questionnaires evaluate Pixie and compare it against text-based passwords on perceived security, ease of use, memorability and speed dimensions.

In addition, at the end of session 3, we have used "emocards" [23] to evaluate the emotional responses of users toward Pixie and text password authentication. Emocards are 16 cartoon faces, each representing one of 8 distinct recognizable facial expression (1 per gender). Emocards assist users to non-verbally express their emotions about products, in terms of pleasantness (pleasant, neutral, unpleasant) and arousal (calm, average, excited), two commonly accepted dimensions of emotion responses [61].

Participant dropout. The participant drop from session 1 to session 3 is not due to a dislike of Pixie. To conclude this, we have compared the distributions of the answers of the 21 participants who dropped and of the 21 participants who stayed until session 3, on their overall impression of Pixie and their willingness to adopt it. Both questions were rated on a Likert scale. The Mann-Whitney test shows that the difference between the two populations is not statistically significant ($p = 0.7532$ for the first question, and $p = 0.0701$ for the second question at $\alpha = 0.05$). The participant drop can be due to the difficulty of scheduling 3 sessions across 8 days, at the end of the semester.

Ethical considerations. We have worked with our university Institutional Review Board to ensure an ethical interaction with the participants during the user study. We have asked the participants to avoid choosing sensitive trinkets. The entire experiments took around 40 minutes per participant. We compensated each participant with a $5 gift card.

6.2 Results

Pixie is a novel authentication solution. Thus, we first present insights from its use across the 3 sessions, with a focus on discoverability. We then detail Pixie's observed memorability and performance, as well as the participant perception and emotional responses. All the statistical tests performed in this section used a significance level of $\alpha = 0.05$.

6.2.1 User Experience. We now detail the user experience across the 3 sessions.

Session 1: discoverability. Without previous knowledge of Pixie, 86% of the participants (36) were able to correctly set up their trinkets. Therefore, Pixie's Failure to Enroll (FTE) rate is 14%. From the 14% (6) participants who failed to enroll, 3 did not notice that the 3 trinket photos had to be of the same object, captured from similar angles. While Pixie provides a tooltip on the trinket capture button that guides the user to take another picture of the trinket when the app is used for the first time (see Figure 1(a)), these 3 participants took random pictures from different objects in the lab. These participants also did not understand the meaning of several words, as English was their second language:

> [P20]: *"Include one page saying what the trinket is. Like [sic], you can say that trinket is an object that you will be using to sign in to your account".*
>
> [P21]: *"I don't understand what plain texture means".*

In all 3 cases, the Pixie prefilters identified the issue correctly. The other 3 unsuccessful participants chose trinkets with a plain texture (e.g., palm of hand, pencil, objects with plain black surface) that

generated errors. They either dismissed error messages quickly or were not sure what to choose as a trinket to eliminate the errors.

Subsequently, 3 other participants were unable to perform the trinket verification step within 3 trials. This occurred due to (i) bad lighting conditions around the trinket, (ii) the participant forgetting the trinket angle, or (iii) a texture-less (plain) trinket. While the Chi-square test did not identify significant differences in the error rates caused by any of the aforementioned circumstances ($p > 0.05$), this could be because of the limited number of samples.

Session 1: Training. All the participants were able to set up a trinket successfully, reducing the FTE rate of Pixie from 14% in the discoverability step to 0%. All the participants then tested their trinkets within 4 trials (M = 1.29 trials, Std = 0.6): 76% of the participants were able to login from the first trial. The other 24% had lighting related difficulties (e.g., the trinket reflected the light, or was in the shadow). Only one participant required 4 trials.

Session 1: Repeatability. All the participants except one, were able to successfully complete this step within 3 trails (M = 1.29trials, Std = 0.6). One participant required 4 trials.

Sessions 2 and 3. In session 2, 84% of the participants were able to login from the first trial, 13% logged in within 2-3 trials and only one participant needed 6 trials (M = 1.35 trials, Std = 1.02). 2 participants did not carry their trinkets and had to reset them. In session 3, 81% of the participants were able to login from the first trial and all the other participants were able to login within 2-3 trials (M = 1.20 trials, Std = 0.40).

6.2.2 Participant Performance. To measure participant performance we use *success rate* [14], defined as the number of successful attempts to the total number of attempts. In order to compare the success rate of participants for text-based passwords and Pixie, we analyzed the data from either of the login recalls of each session. We only consider successful Pixie authentication sessions within 3 trials (see § 6.2.1). This is similar to MyFIU, where the participants need to reset their passwords after 3 unsuccessful trials. The success rate of Pixie improves from session 1 (82.00%) to session 2 (83.33%) and session 3 (84.00%). Throughout all the 3 sessions, the Pixie success rate for successful authentication sessions is slightly lower than the success rate for the text-based password in session 1 (88.10%). This is not surprising, given the significantly lower number of practice opportunities for Pixie, compared to the ubiquitous text passwords. However, the Chi-square test revealed no significant difference between the success rate for Pixie and text password in session 1 ($\chi^2(1) = 0.506, p = 0.48$). Similarly, the Wilcoxon-Mann-Whitney test found no significant difference in terms of the number of attempts for a successful login for Pixie within different sessions, and between Pixie and text-based password in session 1.

6.2.3 Memorability. During session 2, 96% of the participants (all except one) were able to remember their trinkets. 2 participants did not immediately recall the part of the trinket they used to authenticate, but they figured it out in the 3rd attempt. These 2 participants were able to login in the first attempt in the 3rd session. 2 participants did not carry their trinkets and had to reset them in session 2. During session 3, all the participants were able to remember their trinkets. We contrast these results with the memorability of text passwords: 5 participants did not remember their MyFIU password and had to reset it in the first session. This is consistent with previous findings: Wiedenbeck et al. [85]) report that more than 17% of text-based passwords are forgotten in one week.

6.2.4 User Entry Time. We have measured the *user entry time*, the interval from the moment when a user starts Pixie and when Pixie submits the captured photo to the authentication module. Figure 9 shows the box plot of the user entry time for Pixie in different sessions vs. the time for text passwords, during session 1. The shortest authentication session was 3.01s and the longest session was 70.51s for

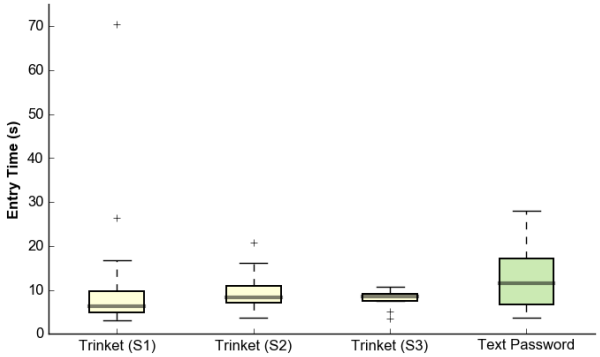


Fig. 9. Box plot for entry time of Pixie across 3 sessions vs. text password in session 1. The Wilcoxon-Mann-Whitney test revealed that **Pixie's entry time in each session was significantly less than the entry time for text passwords**. For a single participant, the Pixie entry time was 70.51s during session 1.

Pixie. The average entry time improves from session 1 (M=9.71s, Std=11.42s, Mdn=6.24s), to session 2 (M=9.71s, Std=4.66s, Mdn=8.32s) and session 3 (M=7.99s, Std=2.26s, Mdn=8.51s). However, Wilcoxon-Mann-Whitney tests did not reveal any statistically significant differences between the Pixie user entry time across the 3 sessions. We expect however that additional practice can further improve Pixie's entry time.

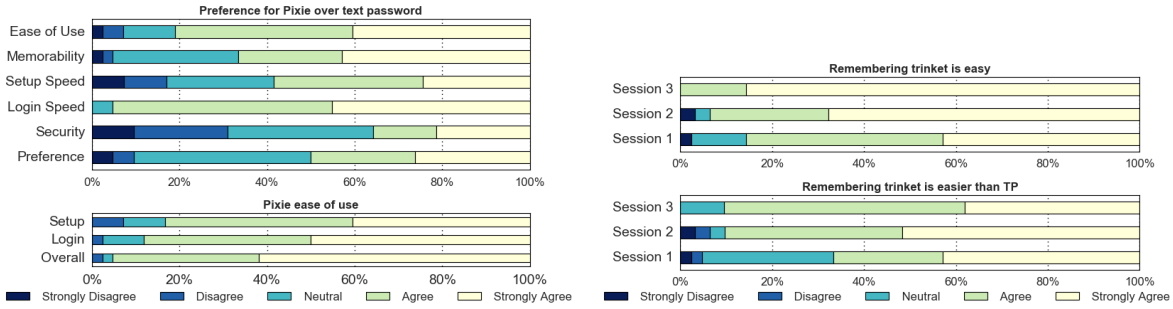
Moreover, a Wilcoxon-Mann-Whitney test revealed that the entry time for Pixie was significantly less than the entry time for text passwords in session 1 ($W = 845.0, p = 0.000$). We emphasize that in contrast to text passwords, Pixie participants did not have the opportunity to practice beyond the steps of the above procedure.

Table 1 compares the entry time for Pixie and other authentication solutions based on biometrics or text and graphical passwords. Although Pixie's entry time is higher compared to solutions based on face or voice, it compares well to several other solutions. For instance, Shay et al. [70] report an entry time of 11.6-16.2 for text passwords. MyFIU passwords are similar to the comp8 category in [70] (at least 8 characters, and include a lowercase English letter, uppercase English letter, and digit) for which [70] report a median entry time of 13.2s. The additional safeguards of Boehm et al.'s [8] face and eyes based biometric solution result in an entry time of 20-40s. Chiasson et al. [14] report an entry time of about 15s for Passpoints. Trewin et al. [76] reported an entry time of 8.1s for gesture (stroke) based biometric. The eye tracking solution of Liu et al. [44] requires 9.6s and the audio or haptic based solution of Bianchi et al. [6] requires $10.8 - 20.1$s.

In addition, we evaluated the processing overhead of Pixie: the time required to decide if a candidate image matches the reference set. The average processing overhead of Pixie on the HTC One smartphone over 94 successful authentication trials is 0.5 seconds.

6.2.5 Perception. We asked the participants to express their perception about Pixie and text passwords by providing answers to a set of questions in a 5-point Likert scale (from strongly agree to strongly disagree). In the following we presents the participants response.

At the end of session 1, 81% of the participants said overall, Pixie is easier to use than text-based passwords (Figure 10(a) (top)). 83% and 86% of the participants agree or strongly agree that the trinket setup and login steps are easy (Figure 10(a) (bottom)). 95% of participants agree or strongly agree that overall, Pixie is easy to use.



(a) (b)

Fig. 10. (a) Results at the end of session 1. (a - top) Perceived performance of Pixie compared to text passwords. Pixie dominates on ease of use, memorability and speed dimensions. (b - bottom) Pixie ease of use: 95% of participants agreed that Pixie is easy to use. (b - top) Pixie perceived memorability. 86% of participants agree that the trinkets are easy to remember after session 1, but reach consensus after session 3. (b - bottom) Perceived memorability of Pixie vs. text passwords (TP). No participant believes text passwords are more memorable after session 3.

Furthermore, 86% of the participants agree or strongly agree that trinkets are easy to remember, see Figure 10(b) (top). 67% of the participants agree that trinkets are easier to remember than passwords, while only 5% of the participants believe the opposite, see Figure 10(a) (top) and Figure 10(b) (bottom). These results improve in sessions 2 and 3. At the end of session 3, all the participants agree that trinkets are easy to remember (Figure 10(b) (top)): 12 participants changed their opinion in favor of Pixie's memorability. No participants believe that text passwords are easier to remember than trinkets, see Figure 10(b) (bottom). A two-sample proportion test revealed that the proportion of the participants who think Pixie is memorable, significantly increases from session 1 to session 2 and 3 ($Z = 2.36$, $p = 0.009$ and $Z = 2.05$, $p = 0.020$).

36% of the participants believe that Pixie is more secure than text passwords, and 31% of the participants believe that passwords are more secure (Figure 10(a) (left)). Several participants felt strongly about the security of Pixie, e.g.,:

> [P27] *"This method is even more secure than text-based passwords, because even if someone sees me during the password entry, he wouldn't know what part of the object I have selected as my trinket and cannot easily figure it out".*

68% of participants agree or strongly agree that the trinket based login is fast. 74% of participants agree or strongly agree that the trinket setup step is fast. 95% and 59% of the participants agree that Pixie's login and trinket setup steps are faster compared to the corresponding text password operations. (Figure 10(a) (top)). 50% of the participants say that they prefer trinkets over text passwords (Figure 10(a) (top, bottom bar)).

When asked if they would use trinket based authentication in real life 26% of participants said that they would use Pixie for most of their accounts, 36% would use it for at least some of their accounts, and 36% would consider using it. Only 2% of the participants (1) said that they would not use it. Several participants felt strongly about adopting Pixie:

Table 13. Confidence interval for the proportion of "agreement" answers to usability and security questions comparing Pixie and text-based authentication. Pixie is perceived to be easier to use, more memorable and faster than text passwords. Pixie's perceived advantage in ease of use, memorability, and login speed is not due to random choice.

Question	Sample proportion	95% CI	p
Easier to use	80.95	(65.88, 91.39)	0.000*
More memorable	66.67	(50.45, 80.43)	0.044*
Faster Login	95.24	(83.83, 99.41)	0.000*
Faster Setup	58.54	(40.96, 72.27)	0.441
More secure	35.71	(21.55, 51.97)	0.088

* Statistically significant result at $\alpha = 0.05$.

[P18]: *"Why isn't [Pixie] integrated with the original MyFIU mobile application as another option for signing to my account?"*.

[P40]: *"I always forget my passwords [...] I always store them in my browser. I would definitely use Pixie if it is available"*.

[P27]: *"I think this is a good method because I usually forget my passwords for my accounts"*.

While we did not include survey questions on trinket availability, one participant asked:

[P8]: *"What if I do not wear the same watch everyday?"*.

Other participants suggested to use multiple trinkets to ensure trinket availability:

[P21]: *"That would be good if we could set multiple trinkets and use any of them to authenticate"*.

Statistical analysis. To differentiate true choice from random chance, we combine the strongly agree and agree answers into an "agreement" answer, and the strongly disagree and disagree answers into a "disagreement" answer. We then use a one-sample binomial test with a confidence interval in order to test whether the proportion of agreement of the participants with a statement is sufficiently different from a random choice (50%). Table 13 presents this result for the proportion of "agreement" answers to each question. Pixie is perceived easier, more memorable and faster than text passwords for login and the perceived advantage is not due to random choice. However, the participants did not perceive a significant difference in the setup speed and the security of Pixie over text passwords.

Analysis of User Feedback. Table 14 shows that the general preference of Pixie over text passwords significantly correlates positively with its preference on ease of use, memorability and security and speed dimensions. The preference over text passwords is also significantly correlated with overall perception of trinket memorability and willingness to adopt Pixie. Interestingly, we observed a significant correlation between preference over text passwords on security and the participant feeling of owning a unique trinket ($\tau = 0.36, p = 0.005$).

The participant willingness to use Pixie also correlates positively with perceived memorability ($\tau_b = 0.29$), perceived ease of use ($\tau_b = 0.28$), general preference over text passwords ($\tau_b = 0.32$), preference over text passwords on security ($\tau_b = 0.28$), and preference on ease of use ($\tau_b = 0.04$). We observe a negative correlation between the willingness to use Pixie and the number of login attempts ($\tau_b = -0.16$), highlighting the impact of unsuccessful logins. However, the correlations are not statistically significant.

Table 14. Kendall's Tau-b test shows significant positive correlation between preference of Pixie vs. text passwords, and its preference in terms of ease of use, memorability, security, faster setup and login time. Preference over text passwords is also significantly correlated with the overall memorability of the trinket and willingness to adopt Pixie.

Prefer Pixie over text passwords	τ_b	p
Easier to use	0.60	0.000*
More memorable	0.54	0.000*
More secure	0.38	0.003*
Faster Setup	0.46	0.000*
Faster Login	0.48	0.000*
Pixie Memorability	0.40	0.003*
Willingness to Use Pixie	0.60	0.000*

* Indicates a statistically significant correlation at $\alpha = 0.05$.

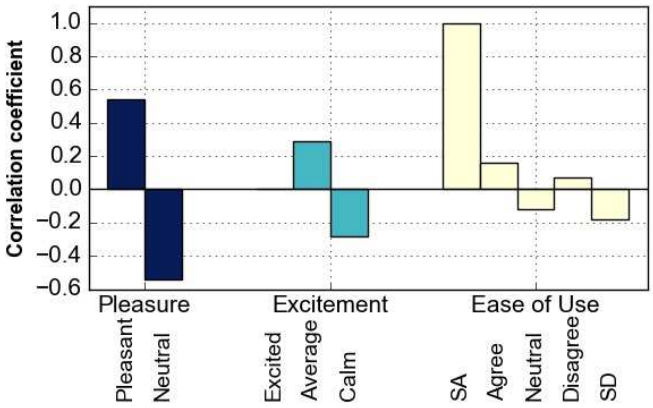


Fig. 11. Kendall's Tau-b correlations between willingness to use, emotional responses (pleasure and excitement), and ease of use (SA/SD = Strongly Agree/Disagree), during session 3. No participant rated Pixie as unpleasant. Willingness to use correlates positively with pleasant and average levels, as well as with agreement with ease of use.

6.2.6 Emotional Response. The emocard experiment revealed that Pixie generates only positive emotions: 81% of the participants reported a "pleasant", and 19% reported a "neutral" experience. In addition, 47% of the participants were "calm", 34% were "average" and 19% were "excited". In contrast to Pixie, only 5% of the participants (1) reported a "pleasant" level for text passwords, while 57% reported "unpleasant" and 38% reported "neutral" levels. A one-sided test of the difference of proportions revealed that the proportion of the participants who perceived Pixie as pleasant was significantly larger than the proportion of the participants who perceived text passwords as pleasant ($Z = 4.01$, $p = 0.000$).

The Kendall's Tau-b correlations plotted in Figure 11 shows that the participant reports of willingness to use Pixie correlate positively with levels of pleasure and excitement, as well as Pixie's perceived ease of use. While 4 participants reported excitement for Pixie's novelty, functionality and performance, we observe no correlation between "excited" levels and willingness to use. This is a positive finding, as authentication solutions should not generate high arousal levels.

6.2.7 Trinket Choice. We manually analyzed the trinket images captured by the participants in the first session (42 trinkets) and those captured by the participants who reset their trinket in session 2 (2

Table 15. Trinket choice: object types chosen by participants, along with the number of unique objects belonging to each category and number of unique trinket choice (object + angle) in the study. The gum pack and watch (used in the training step and on-screen instructions) are the types most frequently used by the participants. All the captured watch trinkets are unique.

Object type	# of unique objects	# of participants	# of unique trinkets
Gum pack	3	16	8
Watch	6	6	6
Mug	3	3	3
Logo	2	2	2
Keychain	2	2	2
Car remote control key	2	2	2
Sunglasses	2	2	2
A piece of puzzle	1	1	1
Shoe	1	1	1
Kohl container	1	1	1
Backpack pin	1	1	1
Hair clip	1	1	1
Cigarette box	1	1	1
Match box	1	1	1
Water Bottle	1	1	1
iphone menu	1	1	1
University ID card	1	1	1
Tattoo	1	1	1
Total	**31**	**44**	**36**

trinkets). We allowed the participants to pick any nearby object as a trinket. The 42 participants picked a total of 36 unique trinkets, from 31 unique objects of 18 types, chosen from among participant owned objects and lab objects. The gum pack and watch were the most frequently chosen object types. However, all the 6 watch trinkets were different, and the 16 participants who chose a gum pack have captured 8 unique trinket images (object + angle combination).

In the discoverability step, 8 participants used their watches as trinkets. We did not observe a significant difference in user choice of trinket between the discoverability and training steps: 18 participants used the same trinket in the discoverability and training steps. 8 participant chose their trinket to be their watches. The other trinket categories chose by participants that are not among those in Table 15 include: pen/pencil, book and computer mouse.

We have used the images captured by the participants to "brute force" the reference sets of each participant. We removed 8 reference sets as they were identical (the top view of the same gum pack). This has produced a single "success" event, for the two participants who chose the same side of the same gum pack, with very similar angles. As we described in § 6.2.5, the participant preference of Pixie over text passwords on security correlates significantly with the participant feeling of owning a unique trinket. We did not observe a statistically significant difference between the feeling of owning a unique trinket and participants gender.

7 DISCUSSION AND LIMITATIONS

Authentication speed. Our user study shows that Pixie's authentication speed in session 1 is 25% faster than well rehearsed text passwords and improves through even mild repetition. However, Pixie's entry time is longer than the reported entry time for face based authentication solutions (see Table 1). This may be due to either the novelty of Pixie or the way the images are captured, i.e. using the back, not the front camera for capturing trinket images.

Secure image storage and processing. The storage and processing of the trinket images needs to be performed securely. While outside the focus of this paper, we briefly discuss and compare trinket image storage and processing solutions that are performed on the remote service vs. the user's authentication device. A remote server based solution trivially protects against an adversary that captures the authentication device, as the device does not store or process sensitive user information. The image matching process is also faster on a server than on a mobile device (see § 5.2). The drawbacks are the overhead of transmitting candidate images over the cellular network, and the imposition on users to register a different reference image set for each remote service.

The authentication device based solution can easily associate the reference images with the user's authentication credentials (e.g. OAuth [18]) for multiple remote services. However, since an attacker can capture and thus access the storage of the mobile device, reference images cannot be stored or processed in cleartext. The storage and processing of reference images can however be secured through hardware-level protection, e.g., TrustZone [77], or by using privacy preserving image feature extraction solutions that work in the encrypted domain, e.g., [36, 57, 81].

Deployability. Pixie is well suited for OAuth [18] authorization to access remote services from the mobile device: Pixie authenticates the user to the app on the mobile device, which can then proceed with the OAuth protocol with the remote server.

Default authentication. If the trinket based authentication fails a number of times (due to e.g., forgotten trinket, poor lighting conditions, unsteady hand), the user is prompted to use the default authentication solution, e.g., text password.

Strong passwords. Popular and ubiquitously available trinkets (e.g., iWatch, Coke can) should not be chosen as trinkets, as an adversary can easily predict and replicate them. To address this problem, Pixie can store a dataset of popular trinket images, then, during the trinket setup process, reject reference sets that match popular trinkets.

Defense against brute force attacks. The brute force attacks of § 3.3 can be made harder to launch through video "liveness" verifications, e.g., [58]: capture both video and accelerometer streams while the user shoots the trinket, then use video liveness checks to verify the consistency between the movements extracted from the two streams. The lack of such streams or their inconsistency can indicate a brute force attack.

The user study. The study presented in this work was the first attempt to quantify the usability aspects of an authentication solution based on trinkets. We performed the user study in a lab setting. We were able to recruit only 42 participants, of which half did not stay until the last session. As we wanted to ensure a consistent and familiar login procedure to remote services for the participants, we only recruited students from the university who had access to myFIU, FIU's login portal. While the population of the study is not fully representative of the users who would use the system, we believe that the participants had no unfair advantage when compared to other social groups of similar age in performance: the participants in our study achieved text password authentication times on par with previously reported results (see § 6.2.4).

Pixie works by extracting invariant keypoints from the captured images, using keypoint extraction algorithms (e.g. SURF [3] and ORB [60]). These algorithms are not capable of extracting keypoints from images of object with constant shade. We attempted to address this issue by providing actionable feedback to users, and guiding them toward choosing visually complex trinkets. In addition, to ensure Pixie is able to identify the trinket images even when captured in slightly different circumstances and to lower the false reject rate, we required the users to enter 3 trinket images in the registration phase. This may partially explain why the participants in our study did not perceive Pixie as significantly faster than text passwords for the registration phase.

During the discoverability step, we observed that several participants had difficulties in understanding the in-app instructions on how to use Pixie. Similar problems have been reported for other authentication mechanisms. For instance, Bhagavatula et al. [4] reported that 7 out of 10 participants found understanding on-screen instructions difficult for iPhone fingerprint authentication. They recommend to provide clearer instructions (e.g. through a demo video) on what the users need to do. We posit that explaining the meaning of trinkets will help users during the registration phase and improve the discoverability rate of Pixie.

The consent form that we read and the participants signed prior to the study, emphasizes that the focus of the study is on the usability aspects of a new authentication mechanism. We observed that some of the participants might have selected trinkets without concerns over security during the study. We did not guide the participants towards choosing specific trinkets, as we intended to observe the personal or lab objects chosen by the participants. Nevertheless, we observed that the participants preference of Pixie over text passwords on security correlates significantly with the participants feeling of owning a unique trinket. This suggests that the participants could corroborate the relationship between unique trinkets and higher level of security.

The trinkets used to walk the participants through Pixie (i.e., gum pack) and the in-app user guide of Pixie (i.e., watch), appear to influence the participant trinket selection in the first session: in the discoverability step, 8 participant chose their watches as trinkets. In addition, during session 1, 9 participants chose the same gum pack as used in the Pixie walk-through without even trying a different angle, and 5 participants used their watches as trinkets. Further studies are required to understand whether other means of communicating the goals of Pixie (e.g. using a short video that guides the user on how to choose secure and unique trinkets) can reduce this bias.

Further, although 50% of the participants said they prefer Pixie over text passwords, 40% percent of the participants were undecided. This may be due to the limited experience of the participants with Pixie. In addition, 62% participants said they would use Pixie in real life. We did not observe a statistically significant correlation between being excited about using Pixie, that could be due to the novelty of the method, and willingness to use it. However, future studies are required to understand in what scenarios and situations the users are willing to adopt trinket based authentication or prefer it over text passwords.

If we were to redo the study, we would split the Likert scale questions comparing Pixie with text passwords into 2 questions asking the participants to rate any of them in terms of usability and security. In addition, we would ask the participants to justify their answers about perceived usability and security in the form of open ended questions in the post study interview.

Real world limitations. A comprehensive study similar to the studies conducted for biometric based solutions (e.g. [4]) may help identify potential limitations of Pixie in different situations. We discuss now two such situations.

- *Insufficient light.* In our studies, we observed that Pixie has problems with insufficient lighting. This is likely a problem shared also by face based authentication solutions. We took steps to partially address this problem, by designing image filters to identify the problematic images and provide users with actionable feedback. Note that low light photography is one of the major differentiators among mobile phone manufacturers. Newer devices are equipped with wider apertures which capture more light and optical stabilization which allow for longer exposures and thereby taking better low light photos. As mobile device cameras get more capable in capturing better low light photos over time, we expect it to be less of a problem for camera based authentication methods as well. Further, while we did not test this in our studies, we conjecture that using the camera's flash light to illuminate the trinket could also help address this problem. We leave it

for future work to investigate better solutions, e.g., that leverage the ability of mobile device cameras to capture infrared light, to handle the case of reflective trinkets.

- *Unstable capture conditions.* Pixie may be harder to use in certain circumstances, e.g., while the user is walking or in public transportation, as movements might affect the quality of snapped photos. The ease of using Pixie in such scenarios is likely to depend on the trinket object type, as for snapping a trinket image, the user needs one or both hands. Pixie has specialized filters to identify blurry images. However, a thorough evaluation of the filters in such scenarios will help identify the filter parameters that maximize usability.

Changing and forgetting trinkets. In contrast to biometric based authentication solutions, Pixie allows the participants to change their trinkets regularly as they change the clothes, accessories and objects that they carry. Despite the obvious security benefits of this property, people may forget to carry their trinkets, impacting Pixie's usability. The simplest solution to this problem is to fall back on default authentication (see above) using a standard approach (e.g., text password) then set a new Pixie trinket. Another solution is to allow Pixie users to have multiple trinkets, the additional trinkets could be chosen among frequently worn outfits or locations visited, alternatively there could be a single backup trinket which is known to be reliably accessible although may not be readily available such as an object kept at home or at work. This however impacts the security of Pixie, as it becomes easier to guess or brute force one of the trinkets.

Another approach is to use Pixie in conjunction with the security token concept, where the token is a printed visual token that displays a pattern (e.g., random art [54]). The user still needs to capture this token using Pixie, and should carry the token at all times, just like for a credit card or mobile device. Using a simple visual token is an alternative to using a QR code as trinket, e.g. [13, 41]). In addition, Pixie does not requre the additional hardware (e.g., magnetic strips and Near-field communication) used in automated access control solutions, and is thus applicable to a wider ranger of mobile and wearable devices.

We note that a survey about the gamut of objects that people carry with them as well as the variability of possession habits would enable further analysis regarding the impact of trinket changes and failure to recall.

Shoulder surfing attack. Choosing and using trinkets in public exposes users to shoulder surfing attacks. In this respect, Pixie is similar to ATM authentication: the user needs to be cognizant of the risks and make sure that there are no people around watching, before setting or using her trinket. We note that while personal privacy during authentication may protect Pixie users from shoulder surfing attacks, this does not hold for biometric authentication alternatives based on face and fingerprints. This is because face and fingerprints are almost public information, accessible to attackers in vulnerable social networks and government datasets. Further, we note that unlike the ATM and biometric authentication scenarios, Pixie allows the user to change her trinket once she is in a more private setting.

We have shown that Pixie is resilient to a shoulder surfing attack flavor, where the adversary learns or guesses the trinket object type and attempts to collect and use images of similar objects to brute force Pixie (see 5.3.2). We note that additional information about the trinket object or the objects owned by the user can increase the adversary chance to launch a successful attack.

In a shoulder surfing attack, the adversary still needs to capture the trinket, or obtain a copy of it to launch a successful attack. Since Pixie authentication requires a simple interaction with the user, it is also possible to combine Pixie with a token (cryptographic key) stored on the mobile device. This approach is similar to the concept of "protocredential" introduced by Corella and Lewison [15]. The combined Pixie and mobile device token authentication would require the user to possess both the particular mobile device that stores the token and the trinket. As an alternative, Pixie can be used in conjunction with

biometric authentication solutions, e.g., [20]: in touchscreen devices, one could use a touch gesture to mark the trinket, as an additional authentication factor.

Field study. We leave for future work a field study of Pixie to investigate the longer term effects of using trinket passwords on user entry times, accuracy and memorability, the factors that impact trinket choice, how users choose and change their trinkets in real life, as well as the potential improvements provided by alternative means of communication of Pixie's goals and functionality (e.g., through short video instead of text). We also leave for future work the investigation of using mental stories to associate trinkets to accounts (e.g., use credit card as trinket for bank account) and reducing the impact of interference [1, 14].

8 CONCLUSIONS

We introduced Pixie, a proof of concept implementation of a trinket based two-factor authentication approach that uses invariant keypoints extracted from images to perform the matching between the candidate and reference images. Pixie only requires a camera, thus applies even to simple, traditional mobile devices as well as resource limited wearable devices such as smartwatch and smartglasses.

We manually captured and collected from public datasets, 40,000 trinket images. We proposed several attacks against Pixie and have shown that Pixie achieved an EER of 1.87% and FAR of 0.02% on 122, 500 authentication attempts and an FAR of less than 0.09% on 14.3 million attack instances generated from the 40,000 images.

We performed an in lab user study to evaluate the usability aspects of Pixie as a novel authentication solution. Our experiments show that Pixie is discoverable: without external help and prior training, 86% and 78% of the participants were able to correctly set a trinket then authenticate with it, respectively. 62% of the participants expressed that they would use Pixie in real life. Pixie simplifies the authentication process: the study shows that trinkets are not only perceived as more memorable than text passwords, but are also easily remembered 3 and 8 days after being set, without any inter-session use. In addition, Pixie's authentication speed in session 1 is 25% faster than well rehearsed text passwords and improves through even mild repetition. We believe that Pixie can complement existing authentication solutions by providing a fast alternative that does not expose sensitive user information.

A promising approach to improve Pixie is to use more advanced image processing techniques, e.g. deep neural networks [74], for image feature extraction and processing. Such techniques may improve Pixie's usability by (i) eliminating the requirement for capturing multiple reference images of the trinket in the registration phase, (ii) increasing the ability to extract features even from images of objects with constant shade, and (iii) further reducing FRRs.

9 ACKNOWLEDGMENTS

We thank the associate editors and reviewers for their excellent feedback and insights. This research was supported in part by NSF grants CNS-1526494, CNS-1527153 and CNS-1422215.

REFERENCES

[1] Mahdi Nasrullah Al-Ameen and Matthew Wright. 2015. Multiple-password interference in the geopass user authentication scheme. In *Proc. Workshop Usable Secur.(USEC)*. 1–6.

[2] Apple Touch Id 2017. Use Touch ID on iPhone and iPad. (2017). https://support.apple.com/en-us/HT201371.

[3] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. 2006. *SURF: Speeded Up Robust Features*. Springer Berlin Heidelberg, Berlin, Heidelberg, 404–417. DOI:https://doi.org/10.1007/11744023_32

[4] Chandrasekhar Bhagavatula, Blase Ur, Kevin Iacovino, Su Mon Kywe, Lorrie Faith Cranor, and Marios Savvides. 2015. Biometric authentication on iphone and android: Usability, perceptions, and influences on adoption. *Proceeding of Usable Security (USEC)* (2015), 1–2.

[5] Andrea Bianchi and Ian Oakley. 2016. Wearable authentication: Trends and opportunities. *it-Information Technology* 58, 5 (2016), 255–262. DOI:https://doi.org/10.1515/itit-2016-0010

[6] Andrea Bianchi, Ian Oakley, and Dong Soo Kwon. 2011. *Spinlock: A Single-Cue Haptic and Audio PIN Input Technique for Authentication.* Springer Berlin Heidelberg, Berlin, Heidelberg, 81–90. DOI: https://doi.org/10.1007/978-3-642-22950-3_9

[7] Robert Biddle, Sonia Chiasson, and P.C. Van Oorschot. 2012. Graphical Passwords: Learning from the First Twelve Years. *ACM Comput. Surv.* 44, 4, Article 19 (Sept. 2012), 41 pages. DOI:https://doi.org/10.1145/2333112.2333114

[8] Arman Boehm, Dongqu Chen, Mario Frank, Ling Huang, Cynthia Kuo, Tihomir Lolic, Ivan Martinovic, and Dawn Song. 2013. SAFE: Secure authentication with Face and Eyes. In *2013 International Conference on Privacy and Security in Mobile Systems (PRISMS)*. 1–8. DOI:https://doi.org/10.1109/PRISMS.2013.6927175

[9] J. Canny. 1986. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8, 6 (Nov 1986), 679–698. DOI:https://doi.org/10.1109/TPAMI.1986.4767851

[10] CaSPR Lab. 2017. Pixie Application. (2017). https://play.google.com/store/apps/details?id=org.image.password.trinket.v1.

[11] CaSPR Lab. 2017. Pixie Data. (2017). https://drive.google.com/drive/folders/0B-qU-nMycga7S1FKbURsU1BlVmc?usp=sharing.

[12] CaSPR Lab. 2017. Pixie Source Code. (2017). https://github.com/casprlab/pixie.

[13] Pan Chan, Tzipora Halevi, and Nasir Memon. 2015. *Glass OTP: Secure and Convenient User Authentication on Google Glass.* Springer Berlin Heidelberg, Berlin, Heidelberg, 298–308. DOI: https://doi.org/10.1007/978-3-662-48051-9_22

[14] Sonia Chiasson, Alain Forget, Elizabeth Stobert, P. C. van Oorschot, and Robert Biddle. 2009. Multiple Password Interference in Text Passwords and Click-based Graphical Passwords. In *Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS '09)*. ACM, New York, NY, USA, 500–511. DOI: https://doi.org/10.1145/1653662.1653722

[15] Francisco Corella and Karen Pomian Lewison. 2015. Protecting credentials against physical capture of a computing device. (April 23 2015). https://www.google.com/patents/US20150113283 US Patent App. 14/588,413.

[16] Mark D. Corner and Brian D. Noble. 2002. Zero-interaction Authentication. In *Proceedings of the 8th Annual International Conference on Mobile Computing and Networking (MobiCom '02)*. ACM, New York, NY, USA, 1–11. DOI: https://doi.org/10.1145/570645.570647

[17] Alexei Czeskis, Michael Dietz, Tadayoshi Kohno, Dan Wallach, and Dirk Balfanz. 2012. Strengthening User Authentication Through Opportunistic Cryptographic Identity Assertions. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security (CCS '12)*. ACM, New York, NY, USA, 404–414. DOI: https://doi.org/10.1145/2382196.2382240

[18] Ed. D. Hardt. 2012. *The OAuth 2.0 Authorization Framework.* RFC 6749. IETF. http://tools.ietf.org/html/rfc6749

[19] Darren Davis, Fabian Monrose, and Michael K. Reiter. 2004. On User Choice in Graphical Password Schemes. In *Proceedings of the 13th Conference on USENIX Security Symposium - Volume 13 (SSYM'04)*. USENIX Association, Berkeley, CA, USA, 11–11. http://dl.acm.org/citation.cfm?id=1251375.1251386

[20] Alexander De Luca, Alina Hang, Frederik Brudy, Christian Lindner, and Heinrich Hussmann. 2012. Touch Me Once and I Know It's You!: Implicit Authentication Based on Touch Screen Patterns. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 987–996. DOI: https://doi.org/10.1145/2207676.2208544

[21] Alexander De Luca, Alina Hang, Emanuel von Zezschwitz, and Heinrich Hussmann. 2015. I Feel Like I'M Taking Selfies All Day!: Towards Understanding Biometric Authentication on Smartphones. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 1411–1414. DOI: https://doi.org/10.1145/2702123.2702141

[22] Alexander De Luca, Marian Harbach, Emanuel von Zezschwitz, Max-Emanuel Maurer, Bernhard Ewald Slawik, Heinrich Hussmann, and Matthew Smith. 2014. Now You See Me, Now You Don'T: Protecting Smartphone Authentication from Shoulder Surfers. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2937–2946. DOI:https://doi.org/10.1145/2556288.2557097

[23] Pieter Desmet, Kees Overbeeke, and Stefan Tax. 2001. Designing Products with Added Emotional Value: Development and Appllcation of an Approach for Research through Design. *The Design Journal* 4, 1 (2001), 32–47. DOI:https://doi.org/10.2752/146069201789378496 arXiv:http://www.tandfonline.com/doi/pdf/10.2752/146069201789378496

[24] Rachna Dhamija and Adrian Perrig. 2000. DéJà Vu: A User Study Using Images for Authentication. In *Proceedings of the 9th Conference on USENIX Security Symposium - Volume 9 (SSYM'00)*. USENIX Association, Berkeley, CA, USA, 4–4. http://dl.acm.org/citation.cfm?id=1251306.1251310

[25] Ben Dodson, Debangsu Sengupta, Dan Boneh, and Monica S Lam. 2010. Secure, consumer-friendly web authentication and payments with a phone. In *Proceedings of Mobile Computing, Applications, and Services*.

[26] Paul Dunphy and Jeff Yan. 2007. Do Background Images Improve "Draw a Secret" Graphical Passwords?. In *Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07)*. ACM, New York, NY, USA, 36–47. DOI:https://doi.org/10.1145/1315245.1315252

[27] Li Fei-Fei, R. Fergus, and P. Perona. 2004. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 Conference on Computer Vision and Pattern Recognition Workshop*. 178–178. DOI:https://doi.org/10.1109/CVPR.2004.109

[28] Nathaniel Wesley Filardo and Giuseppe Ateniese. 2012. *High-Entropy Visual Identification for Touch Screen Devices*. Springer Berlin Heidelberg, Berlin, Heidelberg, 182–198. DOI:https://doi.org/10.1007/978-3-642-29101-2_13

[29] Martin A. Fischler and Robert C. Bolles. 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* 24, 6 (June 1981), 381–395. DOI: https://doi.org/10.1145/358669.358692

[30] H. Gao, X. Guo, X. Chen, L. Wang, and X. Liu. 2008. YAGP: Yet Another Graphical Password Strategy. In *2008 Annual Computer Security Applications Conference (ACSAC)*. 121–129. DOI:https://doi.org/10.1109/ACSAC.2008.19

[31] Jan-Mark Geusebroek, Gertjan J. Burghouts, and Arnold W.M. Smeulders. 2005. The Amsterdam Library of Object Images. *International Journal of Computer Vision* 61, 1 (2005), 103–112. DOI: https://doi.org/10.1023/B:VISI.0000042993.50813.60

[32] Jeff Goldman. 2017. 74 Percent of Organizations Using Two-Factor Authentication Face User Complaints. (Jan. 2017). Retrieved May 5, 2017 from http://www.esecurityplanet.com/network-security/74-percent-of-organizations-using-two-factor-authentication-face-user-complaints.html

[33] Google 2-Step Verification 2017. Google 2-Step Verification User Guide. (2017). https://www.google.com/landing/2step/.

[34] Marian Harbach, Alexander De Luca, and Serge Egelman. 2016. The Anatomy of Smartphone Unlocking: A Field Study of Android Lock Screens. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 4806–4817. DOI:https://doi.org/10.1145/2858036.2858267

[35] Eiji Hayashi, Bryan Pendleton, Fatih Ozenc, and Jason Hong. 2012. WebTicket: Account Management Using Printable Tokens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 997–1006. DOI:https://doi.org/10.1145/2207676.2208545

[36] C. Y. Hsu, C. S. Lu, and S. C. Pei. 2012. Image Feature Extraction in Encrypted Domain With Privacy-Preserving SIFT. *IEEE Transactions on Image Processing* 21, 11 (Nov 2012), 4593–4607. DOI: https://doi.org/10.1109/TIP.2012.2204272

[37] Ian Jermyn, Alain Mayer, Fabian Monrose, Michael K. Reiter, and Aviel D. Rubin. 1999. The Design and Analysis of Graphical Passwords. In *Proceedings of the 8th Conference on USENIX Security Symposium - Volume 8 (SSYM'99)*. USENIX Association, Berkeley, CA, USA, 1–1. http://dl.acm.org/citation.cfm?id=1251421.1251422

[38] A. H. Johnston and G. M. Weiss. 2015. Smartwatch-based biometric gait recognition. In *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. 1–6. DOI: https://doi.org/10.1109/BTAS.2015.7358794

[39] Nikolaos Karapanos, Claudio Marforio, Claudio Soriente, and Srdjan Čapkun. 2015. Sound-proof: Usable Two-factor Authentication Based on Ambient Sound. In *Proceedings of the 24th USENIX Conference on Security Symposium (SEC'15)*. USENIX Association, Berkeley, CA, USA, 483–498. http://dl.acm.org/citation.cfm?id=2831143.2831174

[40] Mohamed Khamis, Florian Alt, Mariam Hassib, Emanuel von Zezschwitz, Regina Hasholzner, and Andreas Bulling. 2016. GazeTouchPass: Multimodal Authentication Using Gaze and Touch on Mobile Devices. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. ACM, New York, NY, USA, 2156–2164. DOI:https://doi.org/10.1145/2851581.2892314

[41] R. Khan, R. Hasan, and J. Xu. 2015. SEPIA: Secure-PIN-Authentication-as-a-Service for ATM Using Mobile and Wearable Devices. In *2015 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering*. 41–50. DOI:https://doi.org/10.1109/MobileCloud.2015.16

[42] Radhesh Krishnan Konoth, Victor van der Veen, and Herbert Bos. 2016. How anywhere computing just killed your phone-based two-factor authentication. In *Proceedings of the 20th International Conference on Financial Cryptography and Data Security*. Barbados.

[43] Wei-Han Lee and Ruby Lee. 2016. Implicit Sensor-based Authentication of Smartphone Users with Smartwatch. In *Proceedings of the Hardware and Architectural Support for Security and Privacy 2016 (HASP 2016)*. ACM, New York, NY, USA, Article 9, 8 pages. DOI:https://doi.org/10.1145/2948618.2948627

[44] Dachuan Liu, Bo Dong, Xing Gao, and Haining Wang. 2015. *Exploiting Eye Tracking for Smartphone Authentication*. Springer International Publishing, Cham, 457–477. DOI:https://doi.org/10.1007/978-3-319-28166-7_22

[45] Päivi Majaranta and Andreas Bulling. 2014. *Eye Tracking and Eye-Based Human–Computer Interaction*. Springer London, London, 39–65. DOI:https://doi.org/10.1007/978-1-4471-6392-3_3

[46] Shrirang Mare, Mary Baker, and Jeremy Gummeson. 2016. A study of authentication in daily life. In *Proceedings of the Symposium On Usable Privacy and Security (SOUPS)*. 6.

[47] J. M. McCune, A. Perrig, and M. K. Reiter. 2005. Seeing-is-believing: using camera phones for human-verifiable authentication. In *2005 IEEE Symposium on Security and Privacy (S P'05)*. 110–124. DOI: https://doi.org/10.1109/SP.2005.19

[48] William Melicher, Darya Kurilova, Sean M. Segreti, Pranshu Kalvani, Richard Shay, Blase Ur, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Michelle L. Mazurek. 2016. Usability and Security of Text Passwords on Mobile Devices. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 527–539. DOI:https://doi.org/10.1145/2858036.2858384

[49] Weizhi Meng, Duncan S Wong, Steven Furnell, and Jianying Zhou. 2015. Surveying the Development of Biometric User Authentication on Mobile Phones. *IEEE Communications Surveys Tutorials* 17, 3 (thirdquarter 2015), 1268–1293. DOI:https://doi.org/10.1109/COMST.2014.2386915

[50] Marius Muja and David G Lowe. 2009. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration.. In *VISAPP (1)*. 331–340.

[51] Douglas L Nelson, Valerie S Reed, and John R Walling. 1976. Pictorial superiority effect. *Journal of Experimental Psychology: Human Learning and Memory* 2, 5 (1976), 523.

[52] Masa Ogata and Michita Imai. 2015. SkinWatch: Skin Gesture Interaction for Smart Watch. In *Proceedings of the 6th Augmented Human International Conference (AH '15)*. ACM, New York, NY, USA, 21–24. DOI: https://doi.org/10.1145/2735711.2735830

[53] Passfaces 2017. Passfaces: Two Factor Authentication for the Enterprise. (2017). http://www.passfaces.com/index.htm.

[54] Adrian Perrig and Dawn Song. 1999. Hash visualization: A new technique to improve real-world security. In *International Workshop on Cryptographic Techniques and E-Commerce*. 131–138.

[55] Andrea Peterson. 2015. OPM says 5.6 million fingerprints stolen in cyberattack, five times as many as previously thought. (September 2015). Retrieved May 5, 2017 from https://www.washingtonpost.com/news/the-switch/wp/2015/09/23/opm-now-says-more-than-five-million-fingerprints-compromised-in-br

[56] Salil Prabhakar, Sharath Pankanti, and Anil K. Jain. 2003. Biometric recognition: security and privacy concerns. *IEEE Security Privacy* 1, 2 (Mar 2003), 33–42. DOI:https://doi.org/10.1109/MSECP.2003.1193209

[57] Zhan Qin, Jingbo Yan, Kui Ren, Chang Wen Chen, and Cong Wang. 2014. Towards Efficient Privacy-preserving Image Feature Extraction in Cloud Computing. In *Proceedings of the 22Nd ACM International Conference on Multimedia (MM '14)*. ACM, New York, NY, USA, 497–506. DOI:https://doi.org/10.1145/2647868.2654941

[58] M. Rahman, U. Topkara, and B. Carbunar. 2016. Movee: Video Liveness Verification for Mobile Devices Using Built-In Motion Sensors. *IEEE Transactions on Mobile Computing* 15, 5 (May 2016), 1197–1210. DOI: https://doi.org/10.1109/TMC.2015.2456904

[59] RSA SecurID 2017. RSA SecurID Suite. (2017). https://www.rsa.com/en-us/products-services/identity-access-management/securid.

[60] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. 2011. ORB: An Efficient Alternative to SIFT or SURF. In *Proceedings of the 2011 International Conference on Computer Vision (ICCV '11)*. IEEE Computer Society, Washington, DC, USA, 2564–2571. DOI:https://doi.org/10.1109/ICCV.2011.6126544

[61] James Russell. 1980. A circumplex model of affect. *Journal of personality and social psychology* 39, 6 (1980). DOI: https://doi.org/10.1017/S0954579405050340

[62] P. Samangouei, V. M. Patel, and R. Chellappa. 2015. Attribute-based continuous user authentication on mobile devices. In *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. 1–8. DOI:https://doi.org/10.1109/BTAS.2015.7358748

[63] Samsung SmartWatch 2017. Samsung Gear SM-V700 Smartwatch. (2017). http://www.samsung.com/uk/consumer/mobile-devices/wearables/gear/SM-V7000ZKABTU/.

[64] Florian Schaub, Marcel Walch, Bastian Könings, and Michael Weber. 2013. Exploring the Design Space of Graphical Passwords on Smartphones. In *Proceedings of the Ninth Symposium on Usable Privacy and Security (SOUPS '13)*. ACM, New York, NY, USA, Article 11, 14 pages. DOI:https://doi.org/10.1145/2501604.2501615

[65] Schlage Keypad Locks 2017. Schlage Keypad Locks User Guide. (2017). http://www.schlage.com/content/dam/sch-us/documents/pdf/installation-manuals/23780042.pdf.

[66] Stefan Schneegass, Frank Steimle, Andreas Bulling, Florian Alt, and Albrecht Schmidt. 2014. SmudgeSafe: Geometric Image Transformations for Smudge-resistant User Authentication. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '14)*. ACM, New York, NY, USA, 775–786. DOI:https://doi.org/10.1145/2632048.2636090

[67] SecuriCode 2017. Ford SecuriCode Keyless Entry Keypad. (2017). http://owner.ford.com/how-tos/vehicle-features/locks-and-security/securicode-keyless-entry-keypad.html.

[68] CaSPR Cyber Security and PRivacy Lab. 2017. Pixie: Two-Factor Camera Based Mobile Authentication. Video. (8 May 2017). Retrieved May 8, 2017 from https://youtu.be/tWepolcXUJg

[69] Ami Sedghi. 2014. Proportion of people working from home reaches record high. (2014). Retrieved May 5, 2017 from https://www.theguardian.com/news/datablog/2014/jun/04/proportion-of-employed-working-from-home-reaches-record-high

[70] Richard Shay, Saranga Komanduri, Adam L. Durity, Phillip (Seyoung) Huh, Michelle L. Mazurek, Sean M. Segreti, Blase Ur, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2014. Can Long Passwords Be Secure and Usable?. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 2927–2936. DOI:https://doi.org/10.1145/2556288.2557377

[71] Maliheh Shirvanian, Stanislaw Jarecki, Nitesh Saxena, and Naveen Nathan. 2014. Two-Factor Authentication Resilient to Server Compromise Using Mix-Bandwidth Devices. In *Proceedings of NDSS*.

[72] 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)*. ACM, New York, NY, USA, 1056–1067. DOI: https://doi.org/10.1145/2976749.2978311

[73] Sony SmartGlasses 2017. Sony SmartEyeglass Developer Edition. (2017). https://developer.sonymobile.com/products/smarteyeglass/.

[74] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2818–2826.

[75] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. 2014. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 1701–1708. DOI:https://doi.org/10.1109/CVPR.2014.220

[76] Shari Trewin, Cal Swart, Larry Koved, Jacquelyn Martino, Kapil Singh, and Shay Ben-David. 2012. Biometric Authentication on a Mobile Device: A Study of User Effort, Error and Task Disruption. In *Proceedings of the 28th Annual Computer Security Applications Conference (ACSAC '12)*. ACM, New York, NY, USA, 159–168. DOI: https://doi.org/10.1145/2420950.2420976

[77] TrustZone 2017. ARM TrustZone. (2017). https://www.arm.com/products/security-on-arm/trustzone/.

[78] Wouter Van Vlaenderen, Jens Brulmans, Jo Vermeulen, and Johannes Schöning. 2015. WatchMe: A Novel Input Method Combining a Smartwatch and Bimanual Interaction. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '15)*. ACM, New York, NY, USA, 2091–2095. DOI:https://doi.org/10.1145/2702613.2732789

[79] VASCO One-Button Authenticators 2017. VASCO DIGIPASS GO Product. (2017). https://www.vasco.com/products/two-factor-authenticators/hardware/one-button/index.html.

[80] Vuzix SmartGlasses 2017. Vuzix M300 Smart Glasses. (2017). https://www.vuzix.com/products/m300-smart-glasses.

[81] Q. Wang, S. Hu, J. Wang, and K. Ren. 2016. Secure Surfing: Privacy-Preserving Speeded-Up Robust Feature Extractor. In *2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS)*. 700–710. DOI: https://doi.org/10.1109/ICDCS.2016.84

[82] Weka 2017. Weka: Data Mining Software in Java. (2017). http://www.cs.waikato.ac.nz/ml/weka/.

[83] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. 2005. Authentication using graphical passwords: effects of tolerance and image choice. In *Proceedings of the Symposium on Usable Privacy and Security*.

[84] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. 2005. PassPoints: Design and longitudinal evaluation of a graphical password system. *International Journal of Human-Computer Studies* 63, 1 (2005), 102–127.

[85] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. 2005. PassPoints: Design and Longitudinal Evaluation of a Graphical Password System. *Int. J. Hum.-Comput. Stud.* 63, 1-2 (July 2005), 102–127. DOI:https://doi.org/10.1016/j.ijhcs.2005.04.010

[86] Anusha Withana, Roshan Peiris, Nipuna Samarasekara, and Suranga Nanayakkara. 2015. zSense: Enabling Shallow Depth Gesture Recognition for Greater Input Expressivity on Smart Wearables. In *Proceedings of the 33rd Annual*

ACM Conference on Human Factors in Computing Systems (CHI '15). ACM, New York, NY, USA, 3661–3670. DOI: https://doi.org/10.1145/2702123.2702371

[87] H. Yoon, S. H. Park, and K. T. Lee. 2015. Exploiting Ambient Light Sensor for Authentication on Wearable Devices. In *2015 Fourth International Conference on Cyber Security, Cyber Warfare, and Digital Forensic (CyberSec)*. 95–100. DOI:https://doi.org/10.1109/CyberSec.2015.27

[88] X. Zhao, T. Feng, W. Shi, and I. A. Kakadiaris. 2014. Mobile User Authentication Using Statistical Touch Dynamics Images. *IEEE Transactions on Information Forensics and Security* 9, 11 (Nov 2014), 1780–1789. DOI: https://doi.org/10.1109/TIFS.2014.2350916