Auth+Track: Enabling Authentication Free Interaction on Smartphone by Continuous User Tracking

Chen Liang[12], Chun Yu[12†], Xiaoying Wei[12], Xuhai Xu[13], Yongquan Hu[1], Yuntao Wang[12], Yuanchun Shi[12]

[1]Department of Computer Science and Technology, Tsinghua University, Beijing, China
[2]Key Laboratory of Pervasive Computing, Ministry of Education, China
[3]Information School | DUB Group, University of Washington, Seattle, U.S.A
{chunyu, yuntaowang, shiyc}@tsinghua.edu.cn,{liang-c19, wei-xy17}@mails.tsinghua.edu.cn,
xuhaixu@uw.edu, im.yongquanhu@gmail.com

ABSTRACT

We propose **Auth+Track**, a novel authentication model that aims to reduce redundant authentication in everyday smartphone usage. By sparse authentication and continuous tracking of the user's status, Auth+Track eliminates the "gap" authentication between fragmented sessions and enables "Authentication Free when User is Around". To instantiate the Auth+Track model, we present **PanoTrack**, a prototype that integrates body and near field hand information for user tracking. We install a fisheye camera on the top of the phone to achieve a panoramic vision that can capture both user's body and on-screen hands. Based on the captured video stream, we develop an algorithm to extract 1) features for user tracking, including body keypoints and their temporal and spatial association, near field hand status, and 2) features for user identity assignment. The results of our user studies validate the feasibility of PanoTrack and demonstrate that Auth+Track not only improves the authentication efficiency but also enhances user experiences with better usability.

CCS CONCEPTS

• **Security and privacy** → **Security services**; • **Human-centered computing** → *Ubiquitous and mobile computing theory, concepts and paradigms.*

KEYWORDS

authentication model, continuous user tracking

ACM Reference Format:
Chen Liang, Chun Yu, Xiaoying Wei, Xuhai Xu, Yongqiang Hu, Yuntao Wang, Yuanchun shi. 2021. Auth+Track: Enabling Authentication Free Interaction on Smartphone by Continuous User Tracking. In *CHI Conference on Human Factors in Computing Systems (CHI '21), May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3411764.3445624

[†] indicates the corresponding author.

CHI '21, May 8–13, 2021, Yokohama, Japan
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8096-6/21/05…$15.00
https://doi.org/10.1145/3411764.3445624

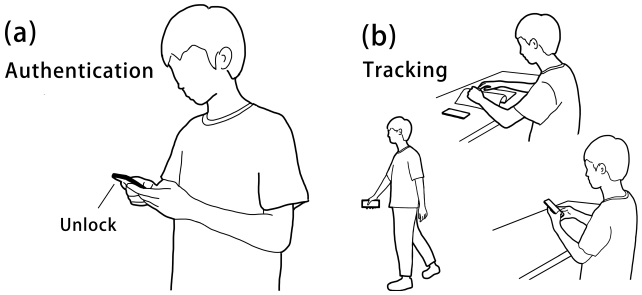


Figure 1: (a) A user is authenticating and the Auth+Track system begins keeping track of the authenticated user. (b) The Auth+Track system continuously tracks the authenticated user in multiple scenes: The user is working while the phone is placed on table; the user is using the phone; the user is gripping the phone and walking. When the user leaves the sensing range of the Auth+Track system, the phone automatically locks.

1 INTRODUCTION

Nowadays, smartphone authentication is indispensable to protect smartphone users' data privacy. However, authentication itself is a tedious and time-consuming process for smartphone users. Numerical and textual password authentication, the original and most commonly adopted authentication form, requires tedious operations and high input delay [21], thus not optimal for mobile usage [58]. Although the procedure in biometric authentication techniques is simplified, users are tired of repeating the authentication again and again [15, 60]. A recent study [23] shows that people spend 2.6 minutes per day in authentication, and authentication procedures are unnecessary for smartphone users in 24.1% cases while [21] demonstrates that people perform 70.3 sessions (39.9 unlocks) per day, taking up 9% of time they use their smartphone. The evidence above reveals that current authentication procedures are not as intelligent as expected. The unnecessary process is perceptible to the user and may annoy the user.

Many solutions have been proposed to reduce the unlocking burden. For example, Google released SmartLock [20] that leverages activity recognition, trusted locations, and trusted devices, to achieve smart authentication, *e.g.*, keeping the phone unlocked when there is a trusted smartwatch nearby. But this method requires additional wearable devices. Moreover, a recent study by

Koushki et al. [44] shows that the misconceptions and difficulty in learning the semantics of multi-modal and context-based unlocking impeded SmartLock techniques from being widely adopted.

An alternative is an implicit authentication, also known as continuous authentication [50]. In previous literature, various implicit authentication methods based on user's behavioral features (e.g., arm movement [35], gait [48], and keystroke actions [42]) have been proposed to reduce the burden of intentional authentication. For example, when someone picks up the phone, their "picking up" behavior is captured by built-in motion sensors and processed by a recognition algorithm. If the movement doesn't match the control pattern of the authenticated user, the authentication system will block them from accessing the phone [35]. However, the shortcoming of these methods is that the authentication accuracy is not high enough for practical usage in daily life [35, 42].

More importantly, most previous work treats each authentication procedure independently, *i.e.*, the current authentication result, either explicit (*e.g.*, entering a password) or implicit (*e.g.*, inferring from arm movement), are not related to the past results [50]. However, when a user interacts with a phone, the usage procedure is continuous and temporally related. There are many cases where repeated authentication is unnecessary and cumbersome when taking historical information into account. For instance, when working on their laptops/PCs, users commonly put their phones besides but need to check incoming notifications frequently. In such a scenario, both explicit and implicit techniques require repeated authentication, which is annoying because the phone should have been staying unlocked when sitting beside users.

To address this problem, we propose **Auth+Track**, a novel authentication model that goes beyond the existing implicit authentication model, and enables "Authentication Free when User is Around" by introducing a continuous user tracking phase to optimize the authentication procedure. Instead of repeating authentication in every session, a user only needs to authenticate once when starting to use their smartphone. The smartphone remains unlocked when the authenticated user is around, as the Auth+Track system automatically keeps track of the user's body movement and accumulate historical tracking records to maintain secure authentication. Once the user leave the scene, or some malign users want to attack the phone (*e.g.*, taking away the device), Auth+Track will lock the phone immediately.

To instantiate Auth+Track authentication model, we present **PanoTrack**, a prototype of Auth+Track based on panoramic scene sensing. In PanoTrack, the status of the body and near field hand is leveraged to control Auth+Track internal state transition logic. We install a fisheye camera on the top of a smartphone (see Figure 4) to achieve a panoramic vision of the surrounding scene, covering users' bodies and on-screen hands. Based on the prototype, we develop an algorithm for authentication. The algorithm first calculates near-field hand status, body keypoints, and the spatial and temporal relation of these keypoints at each frame. Then, it employs the floodfill algorithm to detect the connectivity between the phone and the body. Such a pipeline ensures the tracking robustness when there are multiple persons in the scene.

We conducted two user studies to evaluate the tracking accuracy and the usability of our system. The results show that PanoTrack achieves satisfactory authentication accuracy in real-life scenes. Moreover, users provided positive feedback and rated significantly higher scores of subjective efficiency, performance, and, willing to use. With emerging modern smartphones equipped with power-efficient hardwares[1], realtime, continuous user sensing and tracking on a smartphone is becoming feasible. We envision that Auth+Track can be easily adopted by smartphones in the near future.

Our main contributions are summarized as follows:

1) We propose Auth+Track, a novel authentication model that reduces redundant authentication by continuous user tracking, in order to reduce authentication effort.

2) We present PanoTrack, an instantiation of Auth+Track. With a fisheye camera mounted on a smartphone's top-front, we develop an algorithm to continuously track the user's body movement and the relationship between the phone and the user. Our performance evaluation study demonstrates the good tracking accuracy of PanoTrack.

3) We conduct a second user study to evaluate the usability of PanoTrack in simulated real-life scenarios. Our results show that PanoTrack significantly accelerates the authentication process. Moreover, users provide positive comments on the Auth+Track authentication model.

2 RELATED WORK

We first summarize the existing authentication methods and models on smartphones. We then review vision-based sensing and interaction techniques.

2.1 Smartphone Authentication

Depending on a user's awareness of being authenticated, current authentication methods can be categorized into explicit methods and implicit methods. Supported by these methods, many novel authentication models have been proposed.

2.1.1 Explicit Authentication. Explicit authentication, the original form of authentication, includes password methods (*e.g.*, PINs [46, 59] and graphical patterns [59]) and biometric methods (*e.g.*, fingerprint [25], face [2], and iris [34, 43, 55]). Password authentication methods have become standard on mobile phones. Intuitive numbers or graphical series that people can easily memorize, such as important dates, names, symmetric patterns, are frequently used as passwords. However, such prior information reduces the search space, making password authentication easy to break [13]. In addition to its vulnerability, these methods also ask for user's explicit participation, which is tedious and time-consuming [13]. Biometric authentication methods identify the user based on biometric features, such as fingerprint, face, iris, and voice. Compared with password methods, these methods are more complicated and harder to break. Besides, biometric methods are more efficient, since the user doesn't need to enter a long password sequence or draw a complex pattern. However, the user's awareness of participation still exists, for example, when putting fingers on the fingerprint recognition module, or intentionally facing to the front camera. These acts can bother some smartphone users.

In our work, Auth+Track reduces users' burden by obviating redundant authentication.

[1] For instance, HUAWEI Mate 30, released on September 26th, 2019, has an always-on camera that enables in-air gesture interaction

2.1.2 Implicit Authentication. Implicit authentication, also known as continuous authentication, is a novel concept proposed in recent years and aims to eliminate user's awareness of the tedious authentication procedure [50]. Implicit human behavior features like picking up the phone [35], gait [32, 48], stroke [63], face [39, 54], body posture [47], and voice identification [42] are considered when authenticating a user's identity. Mauro et al. [13] proposed the idea of transparent authentication and modeled people's behavioral feature when answering or placing a call with an accelerator and orientation sensor. Secure Pick Up [35] analyzed the user's arm movement feature when picking up the phone with a smartphone built-in accelerator and gyroscope. DeepAuth [1] illustrated how to do implicit re-authentication in mobile apps based on built-in accelerator and gyroscope data. Papavasileiou et al. [48] developed transparent re-authentication techniques based on gait feature. We point users to a few comprehensive reviews of existing continuous authentication techniques [19, 40, 50, 51]. In these transparent authentication techniques, smartphones serve as a proactive "observer" to monitor a user's behavior and model a user's identity based on implicit behavioral information [51]. However, the major drawback that blocks these methods from commercialization is that these implicit methods are not as reliable as explicit authentication methods like password and fingerprint authentication. As we will show in the paper, our method achieved a more robust results compared to the state-of-the-art methods. Moreover, existing implicit techniques do not leverage historical authentication results effectively, thus still introducing cumbersome authentication process.

2.1.3 Exploration of Novel Authentication Model. Previous work has discussed the balance between usability and security concern [21–23, 30, 41, 52, 60]. Consequently, many authentication models are proposed to optimize traditional authentication processes [5, 26, 27, 53]. CASA [26] introduces an adaptive probability framework to choose appropriate authentication methods based on context dynamically. Progressive authentication [53] divided smartphone usage into 3 security levels (public, private, and confidential) and designed adaptive strategies for each level. SnapApp [5] develops a novel authentication concept by providing a time-constrained quick-access option that bypasses "full access" authentication, which can reduce the authentication workload. All of the work focuses on redesigning authentication processes or the access control logic based on the existing authentication information. But most of them lack additional biometric or behavioral information that helps the system understand a user's behavior to make more intelligent decisions. Although progressive authentication [53] utilizes the sensors to capture certain behavior (accelerometers, light sensors, microphones, and screens), its ability is limited in some common scenes, such as when the smartphone is put on the table (staying still).

In contrast, Auth+Track proposes to continuously track the authenticated user while they are around, which introduces a new "User Around" state in the authentication model. Our prototype PanoTrack can work in a wider range of scenarios.

2.2 Vision based Sensing and Interaction

Vision-based sensing and its corresponding interaction techniques are popular topics in HCI due to an image's compact expression of panoramic scenes. In previous work, different sensing and interaction techniques enabled by different hardware support are widely explored. Different hardware settings, like a RGB camera, a depth camera [9], a RGB camera + prism [62] and a fisheye camera [7, 8, 61], result in different sensing ranges and ability, and thus can capture different semantic information. Sensing information can be categorized into: 1) body-related information [8], such as head, limbs [8], and hand status [7, 62]; 2) object-related information [7]; and 3) the peripheral scene [8].

Based on the extracted semantic features of both the user's body and environment, various active and passive interaction techniques have been developed. Active interaction techniques often integrate sensing information as a new input modality, *e.g.*, finger tracking for on-air cursor control [61, 62], object recognition for body-object interactions [61], and body-and-hand recognition for gesture-based interaction [7, 8]. What interests us is the interaction techniques enabled by these features for passive use. Previous work has tried to incorporate the human body and hand features as contextual information to build intelligent interaction applications. For example, hand gripping status has been used for dynamic layout [11, 37], automatic interface orientation switching [10], and adaptive keyboard decoding [18], while body status and movement have been used for location-based messaging, adaptive interfaces, and followable widgets [61]. Our work is inspired by all these passive sensing designs and is the first to utilize panoramic scene sensing information in authentication state control.

3 AUTH+TRACK

We first illustrate the detailed concept of "Authentication Free when User is Around", a novel sensing goal that leads to a more intelligent authentication process. We then formally introduce **Auth+Track**, a novel authentication model that combines authentication and continuous user tracking, aiming to achieve the goal. Finally, we show how the internal state of Auth+Track transits through a state transition graph.

3.1 Authentication Free when User is Around

Explicit authentication is a tedious and time-consuming process. To illustrate how redundant current smartphone authentication is, we first discuss two common scenes.

Scenario 1: Static Scenario. Alice is working in the office, sitting at her desk, and her smartphone rests beside on the desk. Every time she wants to access her phone, she picks up the phone and deliberately faces the phone to authenticate. She texts a short message and places the phone back on the desk – the phone locks in a minute. The next time she want to access the phone, she need to authenticate again. In this case, repeated authentication is unnecessary because the phone remains around the user and under control.

Scenario 2: Mobile Scenario. Bob is engaged in a multi-round, real-time messaging exchange with his friend while holding the phone in his hand. Every time he sends a message, he waits for the friend's reply. If he waits for more than a minute, the phone locks. He then needs to authenticate again to read new messages and reply. This is bothersome and inefficient because the phone remains on the hand.

In both cases, redundant authentication happens due to the lack of awareness of the user's status. In traditional lock-unlock procedures on smartphones, the phone remains unlocked when it detects touches [20] or the behavior profile that belongs to the owner [50]. However, this criterion is not perfect. Sometimes, a user would like to keep the phone unlocked when the phone is under their control but is not necessarily being used for now.

Therefore, introducing a "user around" state as an active signal could make traditional lock-unlock processes more smooth and natural. When a user is in a "user around" state, they are sufficiently aware of the smartphone [33]. Therefore, putting the user into an "authentication free" status, where the phone keeps unlocked, can help reduce the redundancy of authentication. Based on these conjectures, we define our authentication goal as "*Authentication Free when User is Around*", which guides the exploration of more intelligent lock-unlock procedures.

"User around", meaning user being around and in charge of the device, can be defined based on the smartphone's positional relation with the user. Generally, user-phone relation falls into one of the following scenarios: in-use scenario, put-aside scenario, gripping scenario, pocket scenario, and leaving-away scenario.

"In use" means the user is directly sending instructions to (e.g., touching the screen, clicking a physical button) or receiving information from (e.g., listening to a phone call) the phone. When the user has no intention to interact, while still in charge of the phone, his positional relation with the phone falls into one of the three states: 1) Putting aside. The user puts the phone on a static object around him (within certain distance, e.g., 2m). 2) Gripping. The user is gripping the phone on his hand (while not using it). 3) Pocket. The user put the phone into a carry-on container (e.g., his pocket or bag). "Away" means the user leaves away from the phone for certain distance (e.g., 4m) or is separated with the phone by physical barriers (e.g., a wall), no longer in charge of the phone.

We briefly summarize the status and the possible sensing approaches of each scenario in Table 1. A "user around" state covers put-aside, gripping, and pocket scenarios. We are interested how a user in a "user around" state can be sensed. For "put aside" and "gripping", pixel-wise information captured by phone camera indicating a user's identity, including the face, hands, and body, can be sensed and tracked, while for "pocket", visual evidence is limited to illumination. For "gripping" and "pocket", specific patterns (e.g., gait [1] and a picking-up gesture [35]) from motion data can be used to infer the identity of the user.

Reducing Redundant Authentications by Continuous User Tracking Smartphone usage sessions are fragmented, resulting in numerous "gap authentications". To alleviate unnecessary repetitions, an intuitive approach is to bond the fragmented sessions into a single continuous and coherent session [50]. However, previous work does not cover the "user around" state, which can be leveraged to augment the authentication model. Such a "user around" state can be determined by continuously tracking a user's body movement and hand behavior. Note that before enabling continuous tracking, a "gateway" authentication should be performed to verify the user's identity.

We propose a new model that combines the "gateway" authentication phase and the continuous body tracking phase, called **Auth+Track**. A user only needs to authenticate once when starting to use the smartphone. Then, the phone automatically keeps track of the user's body movement. As long as the user is successfully tracked, the smartphone remains unlocked. We discuss the two phases in detail below.

Scenario	Status	Possible Sensing Approaches
In Use	User on	Camera, motion sensor, and touchscreen
Put Aside	User around	Camera
Gripping	User around	Camera, motion sensor
Pocket	User around	Illumination sensor, motion sensor
Away	User off	–

Table 1: Different scenarios of smartphone usages.

3.1.1 Authentication: A Secure "Gateway". In Auth+Track, the authentication phase serves as a secure gateway to access the phone. Different feed-forward authentication methods, such as password, fingerprint, iris, or face, can be used in this phase. A user's behavioral features vary when they use different feed-forward methods. When using the iris or face, the user's face can be captured. When using a fingerprint or password, the connectivity between the user's body, hand, and phone can be detected. These features serve as the key prerequisite information for the next tracking phase. Auth+Track system first assigns the verified identity to the user and then leverages these features (depending on which method is adopted) to continuously track the user. Accurate user identity assignment is essential because it helps to determine which person is the authenticated user when there are multiple persons in the scene.

3.1.2 Tracking: Continuous Track of User's Behavior. Robust user tracking is crucial to achieving the goal of "Authentication Free when User is Around". After assigning the user's identity, Auth+Track runs a continuous tracking procedure to keep track of the authenticated user's status, which includes the system's tracking information, the range, and the user condition.

These variables change under different sensor solutions. Possible sensor options include capacitive sensors [29], on-device motion sensors [45, 49], on-screen cameras (modality: RGB [61], IR sensors [56], depth sensors [9]; range sensors [62], fisheye cameras [7]), and third-person perspective cameras [12]. Each sensor has its pros and cons. While capacitive sensors and built-in motion sensors are convenient and computationally friendly, they can't sense pixel-wise information or distant behavior. The on-screen camera can capture pixel-wise information, but the sensing range is limited by hardware constraints and the surrounding environment.

We categorize "tracking" into four types according to the range: 1) on-device tracking (*i.e.*, tracking occurring while in contact with the device); 2) near field tracking (<10cm); 3) around-device tracking (0-2m); and 4) full-scene tracking. Although a motion sensor and a capacitive sensor and achieve the first two tracking types respectively, the ranges are too small to be practical. A third perspective camera can capture the full scene, but it is not suitable in mobile cases and has privacy issues. Therefore, we adopt the around-device tracking for continuous user tracking.

3.1.3 Threat Model. When either of the user assignment or user tracking fails, a conservative strategy would be to deactivate Auth+Track, resulting in a normal smartphone usage session without

Auth+Track. An attacker can attempt to cheat the system by deliberately inducing a misassignment or mistracking of the system. Therefore, for any implementation of Auth+Track, robustness in complex environments and the face of deliberate attacks is essential for a successful tracking phase.

3.2 Auth+Track State Transition Graph

In a traditional authentication procedure (Figure 2a), only three states ("User On," "Idle," and "Locked") are recognized. In contrast, Auth+Track separates the "Idle" state into two states – "User Around" and "User Off" – to better represent the full status of users. In Auth+Track, the state of "User Around" is introduced to distinguish a user's status as being around-device or being absent. The state transition graph illustrates the overall workflow of Auth+Track, as shown in Figure 2b. Compared with the traditional authentication model where only touch or "operation" can be recognized for "User On" state, Auth+Track takes advantage of the perception of the surrounding environment and the user's behavior, leading to a more precise representation of a user's status.

4 PANOTRACK: AN INSTANTIATION OF AUTH+TRACK

In this section, we present **PanoTrack**, a prototypical instantiation of the Auth+Track authentication model. In PanoTrack, two categories of information – body movement and near-field hand status – are tracked by an on-device camera, since they are the most informative and identifiable for indicating a users' position and behavior.

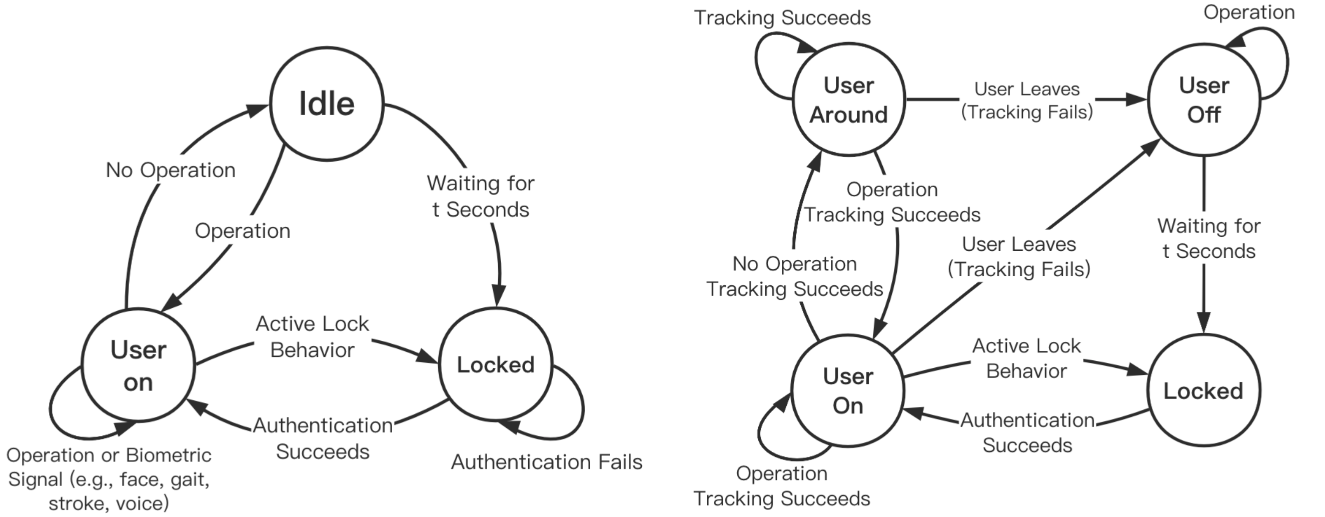
PanoTrack provides a strong sensing capability to capture both the user's body movement and the near-field hand status. After extracting these key features, we design the detailed control logic, *i.e.*, how the PanoTrack system integrates these features to control a phone's state transition logic. Based on how the detected features are used to form the state transition logic of PanoTrack, we propose three strategies: hand-only strategy, body-only strategy, and mixed strategy.

4.1 Hand-Only Strategy

The most straightforward feature to indicate whether the user is in charge of the phone is to detect whether they are gripping, grasping, or touching the phone, which can be captured by the PanoTrack system. An intuitive strategy is that if a "hand-on" signal (gripping, grasping, and touching) is detected, the phone will not automatically lock. However, the main drawback of only using hand information is ambiguity. If there is a release between two "hand-on" signals, we cannot judge whether the two "hand-on" signals are generated by the same user (*i.e.*, the authenticated one). For security reasons, in our design of a hand-only strategy, if there is a release between two "hand-on" signals, the latter one is deactivated. In the PanoTrack hand-only strategy, after authentication, the hand status is continuously tracked. Once the user releases the phone, the tracking is lost. Though this strategy is conservative, it can cover most mobile cases, *e.g.*, using the phone on the road.

4.2 Body-Only Strategy

Compared with near-field hand status, a user's body movement detected by the PanoTrack system is a more comprehensive and reliable feature that indicates whether the user is around. The PanoTrack body-only strategy is designed based on a user's body features, including the body keypoints and their spatial and temporal relationship. After authentication, the PanoTrack system first assigns an identity to the authenticated user, and then continuously tracks the user. When the assignment finishes, the global state of the phone turns from "unauthenticated" to "authenticated". If the



(a) Existing auth model's state transition graph

(b) The Auth+Track model's state transition graph

Figure 2: Authentication Model Comparison. (b) Auth+Track splits the "Idle" state in (a) traditional authentication model into two states – "User Around" and "User Off" – to distinguish a user's status as being around-device or being absent.

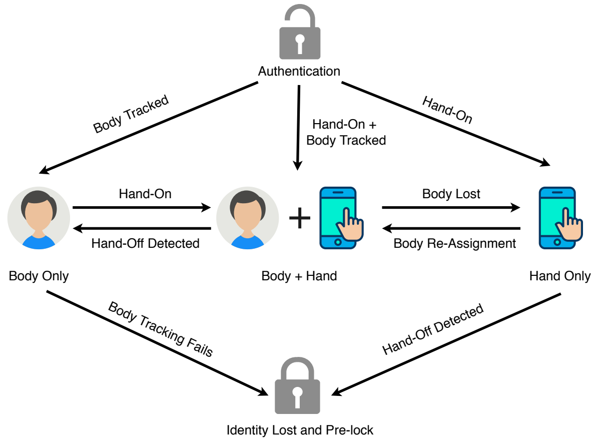


Figure 3: State Transition of PanoTrack Mixed Strategy. After authentication, based on the tracking information, the user falls into one of the three states: body-only, body+hand, and hand-only. The tracking state switches organically based on the relative status between the user and the phone.

tracking succeeds, the global state remains "authenticated". Once the tracking fails (*e.g.*, the user leaves), the global state changes to a "lost" state. When the phone is in a "lost" state, the standard lock procedure initiates, *i.e.*, idling for several seconds, and then locking. Typically, office scenes where smartphone sits still on a desk fit well with this strategy.

4.3 Mixed Strategy

Though the two strategies mentioned above perform well in specific scenes, they have relatively low detection rates in general cases. For example, body tracking in mobile cases quickly fails because of the motion blurring or the absence of a body when a user holds the phone. Similarly, near-field hand status information cannot cope with cases when a hand is absent. To overcome the shortcomings above, we propose a mixed strategy that makes full use of both hand and body features. The key idea to merge a body-only strategy and a hand-only strategy is to reinforce the vitality of one strategy based on the other - making the tracking procedure harder to deactivate. The reinforcement strategy can be divided into "hand-to-body" phase and "body-to-hand" phase.

A "hand-to-body" phase focuses on tracking the scene when a body is absent from the camera's field of view, *e.g.*, a user is gripping the phone and walking. When an activated "hand-on" signal is detected, even if a user's body is untracked or lost, the tracking can be recovered. The authenticated identity is reassigned when the camera system detects the following: 1) a stable center-oriented face close enough to the camera; and 2) connectivity between a user's on-screen hand and the user's body. The reassignment of tracking is reliable and secure in identifying the authenticated user because a "hand-on" signal is tethered to the identity of the hand. And the hand-body connectivity assignment and a center-oriented face assignment assures the relation between the hand in the camera's field of view and the captured body.

"Body-to-hand" phase aims to deal with the situation when there is no hand in a camera's field of view. In the hand-only strategy, if there is a release between two "hand-on" signals, the latter is deactivated, which is not long-lasting. In the "body-to-hand" phase, body identity helps to activate a deactivated "hand-on" signal even if it appears after release. Specifically, when the authenticated user's body is successfully tracked, and then a "hand-on" signal emerges, the connectivity between the smartphone, hand, and body is activated. For example, when an authenticated user grasps his/her phone from the desk, if connectivity is detected, the system status proceeds into "hand mode", and the hand status is continuously tracked. The security of the re-activation is ensured because an "authenticated body" is always connected to an "authenticated hand".

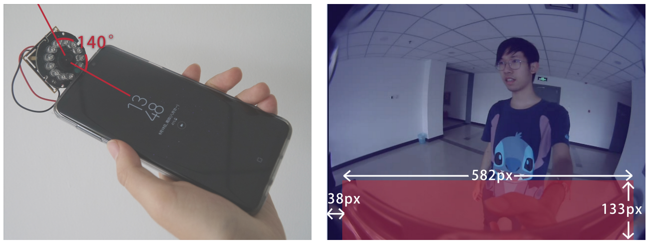


Figure 4: Left: Our hardware prototype. The sensing range is 140° (vertical) × 360° (rotational). Right: Captured image (640 × 480) can be divided into 2 regions: near field region (red) for hand gesture recognition and major region (blue) for user tracking.

Figure 3 shows the relation between hand mode and body mode and the transition logic between them. By applying a mixed strategy, the true detection rate in the wild can be improved while security is ensured.

5 PANOTRACK: HARDWARE

In this section, we describe the detailed hardware design of PanoTrack and explain how to achieve panoramic scene sensing that can capture both a body and an on-screen hand.

5.1 Camera Setting and Sensing Range

To enable panoramic vision that can capture both user-oriented scenes and peripheral scenes, we chose a dual-mode 160° fisheye camera as a proof-of-concept sensor. The camera is fixed to the top of the phone, simulating a potential camera position in future smartphones. To capture clearer on-screen hands and reduce unrelated environment in the peripheral area, we tilted the camera 30° to the bottom. The valid FoV of the fisheye camera is 140°(vertical) × 360°(rotational). To achieve the around-device tracking, the expected sensing radius for is 2m. In Panotrack, the valid sensing radius is 4m to capture a clear human body.

5.2 PanoTrack Hardware Prototype

We used a Samsung Galaxy S9+ smartphone (CPU, RAM: 4GB) with a 5.8-inch touchscreen running Android 7.0. The phone was fixed to a 3D-printed stand with a dual-mode 160-degree fisheye camera on the top. The resolution of the output video stream is 640 × 480, and the FPS is 30.

We implemented PanoTrack's algorithm in a PC server with a GTX 2080Ti NVIDIA GPU with 11GB memory. The algorithm pipeline is implemented in Python and pyTorch. The PanoTrack

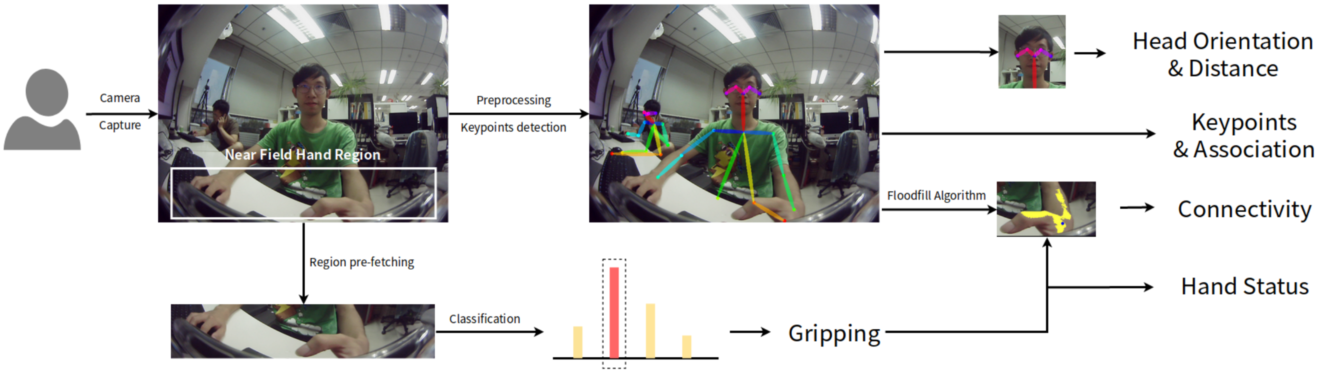


Figure 5: Our algorithm pipeline. First, we extracted near-field hand regions from the original image. Then we extract all the keypoints and their spatial and temporal connections using a CNN-based model. Simultaneously, we predict near-field hand status with a classification model. Finally, we merge body detection results and the hand status classification result to calculate high-level semantic information.

system on the smartphone was implemented on an Android app that runs in the background to control the lock-unlock logic. The fisheye camera was connected to the PC server. The PC server processed the video stream captured by the fisheye camera, analyzed the vital information, and sent commands to the smartphone via WiFi in real-time.

6 PANOTRACK: ALGORITHM

With the video stream captured by the fisheye camera as input, the algorithm aims to output human keypoints, the spatial and temporal relation of the keypoints, and near-field hand status of every frame. PanoTrack can track the user's body movement, detect the near-field hand, and control the authentication status in real-time.

As shown in Figure 5, our algorithm pipeline consists of four parts: preprocessing, continuous body tracking, near-field hand behavior detection, and user-identity assignment. Given an image captured by the fisheye camera, the pipeline first extracted a fixed region in the image for near-field hand detection. Next, a convolutional neural network (CNN) model was applied to acquire all the keypoints and their spatial connections. These keypoints were then associated temporally based on a tracking algorithm. Simultaneously, a near-field hand status was predicted by a classification model. Finally, the detected body keypoints and near-field hand status were used to calculate user identity features including head orientation, head distance and hand connectivity for user identity assignment.

6.1 Preprocessing

Since the smartphone was in a fixed area in the image, the near-field hand gestures (such as gripping, grasping, and touching) were limited to a specific area. For better recognition of near field hand gestures, we pre-fetched the near field region of the smartphone in the image. To empirically determine the area, we collected 100 images of various hand gestures through a pilot study and manually labeled the hand area of each image. The final region boundary is the union of the labeled region (see Figure 4).

6.2 Continuous Body Tracking

The next step of our algorithm pipeline was to track human bodies in the video stream, outputting all the keypoints and their spatial and temporal relationships.

6.2.1 Body Keypoints and Association Detection. We use a CNN-based model to predict the keypoints of body parts in each frame. In our implementation, a pre-trained MobileNet V3 [31] was used as the backbone network for image feature extraction. We followed Cao[6]'s idea to decompose the mission of predicting body keypoints and their connections into predicting confidence maps of every keypoint and part affinity fields (PAFs) of every limb. After confidence prediction and PAF regression, we extracted all the local maxima with high confidence scores as keypoints and applied a biparty graph matching algorithm to predict the optimized joint connection. We fine-tuned the model on the COCO 2017 dataset [38].

6.2.2 Temporal Tracking. By applying the model above, we ascertained the optimized keypoints and skeletons for each independent frame. We proposed a method to associate the detected keypoints in time domain. Given the keypoints and their spacial association, we first organized the result in entity-based perspectives, *i.e.*, each image contains a set of entity-based structures, and each structure stores keypoints, associations, and confidence maps of the entity. For each entity (person) in the current frame, we calculated the probability that it is consistent with another entity in some previous frames by:

$$p = exp(-\frac{\lambda}{|S_{i,k} \bigcap S_{j,l}|} \sum_{n \in S_{i,k} \bigcap S_{j,l}} ||V_{i,k,n} - V_{j,l,n}||^2)$$

where $S_{a,b}$ denotes the valid joint set of the b^{th} person in frame a. $V_{a,b,c}$ denotes the position of the c^{th} joint of the b^{th} person in frame a. λ is a scaling parameter. If probability $p > \epsilon$ and $|i-j| \leq T$, we assigned the identity of the k^{th} person in frame i to the l^{th} person in frame j. We set $\lambda = 10^{-5}$, $\epsilon = 0.9$ and $T = 5$ based on the pilot study.

6.3 Hand Behavior Detection

For hand-behavior detection, we distinguished the following four near-field hand statuses: hand-off, gripping, grasping, and touching. Taking the near-field hand region as input, we used a pre-trained MobileNet V3 [31] model to extract the graphical features of this region and then classify it with a multi-layer perceptron (MLP). We initialize the MobileNet V3 layers with pre-trained parameters and the MLP layers with normalized random parameters [28]. Then we fine-tuned the MLP parameters and the MobileNet V3 parameters with Adam optimizer.

To alleviate the misrecognition caused by blurred frames and to improve prediction confidence and smoothness when determining the label of the current frame, we took the recognition results of the previous 2K (K=2) frames into account. The smoothed label was the voting result of the majority of local predictions of these $2K + 1$ frames.

6.4 User Identity Assignment

The body and hand features acquired from previous steps were further used to extract three high-level semantic features: head orientation, head distance from the camera, and the connectivity between body, hand, and phone.

The head orientation, and distance features were used for immediate user identity assignment after authentication, while the connectivity was used for "hand-to-body" or "body-to-hand" reassignment in the mixed strategy (see Figure 3).

6.4.1 *Head Orientation and Distance.*

The nose, left ear, left eye, right ear, and right eye were the five keypoints that the body keypoint detection model output. These keypoints were used to estimate the user's head orientation along three axes (vertical, horizontal, and rotational) and the distance of the head from the camera.

We define the bias level in 3 axises as:

$$b_{vertical} = 1 + \frac{(X_{lear} - X_{nose}) \cdot (X_{rear} - X_{nose})}{||X_{lear} - X_{nose}|| \cdot ||X_{rear} - X_{nose}||}$$

$$b_{horizontal} = \frac{|||X_{lear} - X_{leye}|| - ||X_{rear} - X_{reye}|||}{|||X_{lear} - X_{leye}|| + ||X_{rear} - X_{reye}|||}$$

$$b_{rotational} = \frac{X'_x + \epsilon}{X'_y + \epsilon}, X' = \frac{X_{leye} + X_{reye}}{2} - X_{nose}$$

where X_* denotes the 2D coordinate of feature point $*$, and X'_x (X'_y) denotes x(y)-coordinate of point X'. ϵ is a smooth factor. If $|b_{verticle}| < \epsilon_1$ and $|b_{horizontal}| < \epsilon_2$ and $|b_{rotational}| < \epsilon_3$, the head is detected as "center-oriented". We set $\epsilon_1 = 0.5$, $\epsilon_2 = 0.1$, and $\epsilon_3 = 0.5$ based on a pilot study. Similarly, we used $||X_{lear} - X_{leye}|| + ||X_{rear} - X_{reye}||$ to estimate the distance of the user's head from the camera.

6.4.2 *Connectivity.*

To detect the connectivity between body, hand, and phone, we developed a strategy based on floodfill algorithm. We choose the wrist keypoints detected by body tracking algorithm as the seeds and apply floodfill algorithm (using OpenCV [4]) in RGB color space, with an empirical lower difference threshold of $L = (15, 15, 15)$ and an upper threshold of $H = (20, 20, 20)$. Starting from the seed points, the algorithm recursively selects the points adjacent to the selected points whose color is within the range of $(C_{selected} - L, C_{selected} + H)$ until no new points can be selected. After running floodfill algorithm, we found the connected areas which include all the wrist keypoints. To judge whether there is connectivity between body, hand, and phone, we detected whether the expanded areas touch or are close enough to the smartphone boundary.

By continuously extracting the head and connectivity features, the algorithm can keep track of the user's identity and the relation between the phone and the user after authentication.

6.5 Efficiency Analysis

Currently, our pipeline runs 23 FPS: 4 ms for image fetching and preprocessing, 19 ms for body keypoint detection, 7 ms for hand status classification, 5 ms for the calculation of temporal features, head features and connectivity, and the rest for real-time visual feedback rendering. Note that our prototypical implementation could be optimized by 1) disabling visual feedback provision and 2) merging the mutual frontend network of the keypoint detection model and the classification model, the theoretical optimal FPS for calculation should be over 30, which is beyond the camera's streaming constraint.

7 STUDY 1: ALGORITHM COMPONENT EVALUATION

We first conducted an user study to evaluate the performance of each component in the PanoTrack algorithm pipeline: 1) body keypoint detection, 2) near field hand status classification, and 3) user identity assignment.

7.1 Participants and Apparatus

We recruited 10 participants (7 males) from the local campus. The average age of all participants was 23.0 (SD=1.34). All participants were familiar with the face authentication method, which we used as the explicit method in this study. The apparatus was the same as described in 5.2 and Figure 4.

7.2 Evaluation Metrics

For body keypoint detection, we measured the recognition accuracy, precision, and recall of users from individual sampled frames at different distances. For near field hand status classification, we measured the accuracy of both 2-class classification (hand-on and hand-off) and 4-class classification (hand-off, grasping, gripping, and operating). For the user identity assignment, we measured the accuracy of the assignment when the users unlocked the phone.

7.3 Data Collection

We collect the data by asking participants to perform specific tasks with video recording turned on. Participants went through a brief introduction of each task and signed the consent form.

First, we collected body data in different distances, which is used to validate the keypoint detection NN model described in Section 6.2.1. Participants were asked to perform free form movement in three distances: 1) $< 1m$, 2) $\approx 2m$, and 3) $\approx 4m$ and we recorded a 60-second video for each participant in each distance.

Then we collected four types of hand status data: 1) hand-off-screen, 2) hand-grasping, 3) hand-gripping, and 4) hand-operating,

Distance	$\leq 1m$	$\approx 2m$	$\approx 4m$
Accuracy	99.4%	99.2%	81.6%
Precision	100.0%	100.0%	97.4%
Recall	99.4%	99.2%	83.4%

Table 2: Body detection accuracy in different distances.

	w Smooth	w/o Smooth
2-Class	99.9%	99.7%
4-Class	97.5%	96.1%

Table 3: Hand status classification accuracy.

to evaluate the hand status classification model described in Section 6.3. Participants were asked to perform each hand gesture and recorded a 60-second video for each gesture.

Finally, we collected data about standard authentication to evaluate our user identity assignment algorithm (Section 6.4). With PanoTrack turned on, each participant was asked to perform 20 authentication attempts (10 with the phone on table, and 10 with the phone in hand) with video recorded. The whole procedure about 20 minutes for each participant.

7.4 Result

We present our results of separate PanoTrack algorithm pipeline components as below.

7.4.1 Body Detection. To evaluate the performance of body detection on individual frames, we uniformly sampled 500 frames (out of $10 \times 60 \times 30 = 18000$ frames) from the captured videos for each distance ($\leq 1m$, $\approx 2m$, and $\approx 4m$). We manually labeled whether the body keypoints were correctly detected in each frame based on the following criterion: A sample was labeled positive when and only when all the body parts in the image and their proper association are correctly detected by the model while yielding no false positives for non-body parts. Table 2 shows the accuracy, along with precision and recall, of our body detection model. The detection accuracy can reach almost 100% (99.4% for $\leq 1m$ and 99.2% $\approx 2m$) when user is within 2 meters, showing the robustness of our body detection model in near range. The results also show high precision in different distances (Even in $\approx 4m$, our model has a high precision of 97.4%), meaning our model would hardly yield a false positive sample (*e.g.*, recognizing an object in the scene as a body part).

7.4.2 Hand Status Classification. We applied a 10-fold cross-validation procedure to evaluate our model for hand status classification. In each fold, one of the participants' was left out as test data. The rest of the data was randomly split into a training set (8 persons) and a validation set (1 person). We trained our model on the training set, chose the epoch that maximized the model's accuracy on the validation set, and tested the model in the test set. The result is shown in Table 3. In 2-class classification (hand-on and hand-off), our model reached an accuracy of 99.72%, while the accuracy of 4-class classification (hand-off, grasping, gripping, and operating) is 96.10%. The error in 4-class classification mainly came from the confusion between gripping and operating data (some of the operating gestures are similar to gripping gestures). By enabling the smoothing procedure, the accuracy was much higher (2-class: 99.94%, 4-class: 97.52%) after a significant portion of blurred frames were filtered.

7.4.3 User Identity Assignment. The overall success rate of in-hand authentication assignment was 100% (100 / 100), while the overall success rate of an on-table authentication assignment was 97% (97 / 100). The result was within our expectation since when users are authenticating with the phone in hand, their face will probably in the center of the image, covering the major area of the image, making the assignment hard to fail. We analyzed the three negative samples and found that two of them failed because of improper distance or orientation: one user was far away from the camera, and the other had a 70° improper orientation. In the third case, the system wrongly assigned another person in the scene because there were two people evenly close to and facing the phone. These three samples could be easily resolved if the face position – detected by the phone's Face ID system when they unlocked the phone – was available to us, since this would provide the initial tracking point and enable PanoTrack to robustly tracking the user body. Due to the hardware limitation, we could not access the Face ID system detection results. But our observation indicates that the negative samples can be avoided once this limitation is relaxed. We will elaborate more on this in the discussion section.

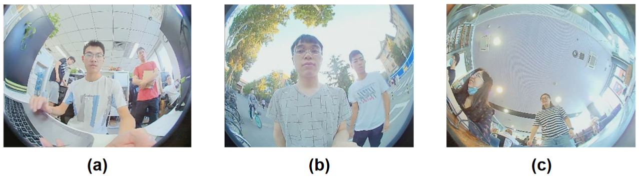


Figure 6: Data sample in different scenes: (a) in the lab/office; (b) in the street; (c) in the cafe.

8 STUDY 2: PERFORMANCE EVALUATION

We conducted a further study to evaluate the overall performance of the PanoTrack system in multiple real-life scenes.

8.1 Participants and Apparatus

We recruited 14 participants (7 males). The average age of Group 2 was 20.6 (SD=1.34). All participants use a touchscreen smartphone on a daily basis for more than four years. Same as Study 1, we used face authentication as the explicit method, with which all participants were all familiar. The apparatus was the same as described in 5.2 and Figure 4.

8.2 Evaluation Metrics

We measured the count of succeeded, failed, and wrong tracking, from which we can calculate precision: the percentage of correct authentication among all authenticated cases, and recall: the percentage of true authentications that are correctly recognized. The precision indicates how robust the system is against attackers and environmental interferences. Lower precision means that there are more cases other people will be mis-recognized as the authenticated user. The recall indicates how much the system can effectively reduce authentication times. Lower recall means that there are more cases the authenticated user is not tracked by the system and requires the user to re-authenticate.

8.3 Data Collection

We were interested in how well PanoTrack tracks the users when they were performing different tasks in different real-life scenarios. So we asked the participants to perform daily tasks and evaluate the performance of PanoTrack in these tasks.

After signing the consent form, participants were instructed to perform 7 tasks close to real-life activities (Table 4) in three common scenes: lab/office, street, and cafe (Figure 6). These scenes cover both indoor and outdoor sites, and the tasks covers daily activities of work, commuting, and entertainment, while covering the most relative status and movements between the user and the phone. Each task took about 20 seconds and was repeated 3 times. Before acting each scene, participants authenticated once to ensure they were successfully tracked at the beginning.

It is worth emphasizing that, to enhance the diversity of our data and make it closer to real-life scenarios, 1) we required the participants to test in a crowded environment and expected that at least two or more disturbers will appear in the background. 2) participants could perform open-form gestures based on their understanding for a same task. For example, for the task 6 (out of camera sight) in Table 4, participants could either walk out of view or put the phone in their pockets. 3) Moreover, our data covered the setting where users switch between scenes organically (*e.g.*, switching from a face-oriented state to a hand-held state by performing task #5).

During the study, tracking succeeded count, tracking failed count and wrong tracking count in these scenes were recorded. All tasks mentioned above were randomized within each session to remove the order effect. The study took about 30 minutes for each participant.

8.4 Result

We present our results of PanoTrack's overall performance in different scenes as below.

8.4.1 Analysis of Tracking Performance in Different Scenes and Tasks. The seven tasks in Table 4 were chosen based on people's daily smartphone usage. We categorized these tasks into two groups based on the phone's status: on a table-like static surface (abbreviated as on table) and in hand. These tasks covered most of phone interaction space. Table 4 summarizes the counts of succeeded, failed, and wrong tracking of different tasks in the different scenes.

Overall, PanoTrack had a high detection precision (99.5%) and recall (94.7%) for around-the-device tasks (1-5) in the three scenes. It achieved a 99.6% tracking precision for on-table tasks (1-2) and 99.4% for in-hand tasks (3-5), meaning the system would robustly track users no matter the phone is on table or in hand.

We found that the tracking recall varied between different tasks. The recall of on-table tasks was 97.2%, while that of in-hand tasks was 93.1%. The difference was mainly caused by the phone's motion. When the user was gripping the phone, the field of view of the camera changed more frequently, leading to motion blurring and body segments being out of sight. We also tested whether the tracking would be correctly disabled once the user left out of camera sight by introducing a "out of camera sight" task (#6 in Table 4). In such task, PanoTrack robustly recognized the user's leaving behavior and had a precision of 98.4%. To further evaluate the tracking robustness in edge cases, we also tested our system in an "leaving away (≈ 4m) and back" task (#7 in Table 4). In this task, the precision was 95.3% and the recall was 83.5%. We found that tracking failures occurred more often in task 7 due to low image resolution, while the system still kept a low mis-tracking rate 4.7%. Other mis-tracking cases were mainly caused by body obstruction from other disturbers and the wrong detection in body keypoints.

We also noticed the performance of PanoTrack varied regarding different scenes. In the street or cafe, the recognition precision is 100% for almost all tasks. PanoTrack performed accurate tracking without yielding any failure or mis-tracking under these two scenes. However, in the lab/office scene, the overall precision was not perfect (although it was also high). The space of the lab/office was small and crowded (Figure 6 (a)), as a result, the influence of disturbers was more significant in such scene (lab/office) than in other scenes, leading to a decrease in tracking precision. In addition, we found the tracking recall of in-the-street scene is significantly lower (90.5%) than that of the others. Analysis on the negative sample revealed the main reason: the body detection accuracy was greatly affected by the overexposure of the camera.

8.4.2 Comparison with Implicit Solutions. State-of-the-art implicit authentication methods used different modalities, such as phone motion (99.2% precision, 92.4% recall [1]), touch stroke (95.0% precision [63]), application usage (97.8% precision [36]), and front camera captured face images (76.15% accuracy [39]). We summarized the behaviors, sensing ranges, sensing modalities, dataset characteristics, and performances (precision, recall, F-1 score, and accuracy) of these methods along with PanoTrack in Table 5 to form a comprehensive comparison and settle the boundary of different methods.

From the table, we found that motion-based (*e.g.*, accelerometer and gyroscope) and touch-based methods [1, 35, 63] have high accuracy. But their sensing range is limited to on-body or on-screen and the user's behavioral constraint is stricter (*e.g.*, walking, picking up the phone, or touching the screen). PanoTrack achieves comparative performance (F-1 Score: 97.0% (PanoTrack) *v.s.* 95.8%(DeepAuth [1]); Accuracy: 94.3% (PanoTrack) *v.s.* 96.3% (SecurePickUp [35])) while having broader sensing range ($\leq 2m$) and more relaxed behavioral restriction. However, PanoTrack is limited when the fisheye camera cannot captured the body or the hand of the user (*e.g.*, in the pocket), where motion-based methods work well. Therefore, one promising future work direction of PanoTrack is to incorporate complementary motion sensors (*e.g.*, IMU) so that its capability could be further enhanced.

Compared with existing vision-based methods [39, 54], PanoTrack achieved significantly better results (F-1 Score: 97.0%(PanoTrack) *v.s.* 82.8%(UMDAA-02 Face [39])) than the baselines. The improvement benefits from: 1) stronger hardware settings (160° FoV *v.s.* normal FoV, 30 FPS *v.s.* 3 FPS) and 2) fine-grained algorithm pipeline design (temporal body+hand tracking, hybrid state control logic (Section 4.3) *v.s.* feature-based classification). Moreover, from the scope of authentication model, existing vision-based methods [39, 54] aim to authenticate, meaning to figure out the right user in a closed set, thus are user-dependent. For example, the performance of attribute-based method [54] will drop as the user set enlarges while the attribute space remains the same. In comparison, PanoTrack aim to track the user after a strong authentication. The tracking phase works regardless of user-related features, thus is user-independent. From this point, PanoTrack is less affected by the size of the user set, thus more realistic in deployment.

Scenes			In the Lab/Office					In the Street					In the Cafe					Overall	
Id	Status	Task Description	S	F	W	Pre	Rec	S	F	W	Pre	Rec	S	F	W	Pre	Rec	Pre	Rec
1	T	Putting aside	42	0	0	1.000	1.000	39	3	0	1.000	0.929	42	0	0	1.000	1.000	1.000	0.976
2	T	Walking around	41	0	1	0.976	1.000	40	2	0	1.000	0.952	40	2	0	1.000	0.952	0.992	0.968
3	H	Using the phone	42	0	0	1.000	1.000	41	1	0	1.000	0.976	41	1	0	1.000	0.976	1.000	0.984
4	H	Gripping while sitting / standing	41	0	1	0.976	1.000	36	6	0	1.000	0.857	41	1	0	1.000	0.976	0.992	0.944
5	H	Gripping while walking	36	5	1	0.973	0.878	34	8	0	1.000	0.810	38	4	0	1.000	0.905	0.991	0.864
6	T / H	Out of Camera Sight	0	40	2	0.952	-	0	42	0	1.000	-	0	42	0	1.000	-	0.984	-
7	T	Leaving away (≈ 4m) and back	35	3	4	0.897	0.921	34	8	0	1.000	0.810	32	9	1	0.970	0.780	0.953	0.835
1-2	T	On-table tasks	83	0	1	0.988	1.000	79	5	0	1.000	0.940	82	2	0	1.000	0.976	0.996	0.972
3-5	H	In-hand tasks	119	5	2	0.983	0.960	111	15	0	1.000	0.881	120	6	0	1.000	0.952	0.994	0.931
1-5	T&H	Around-the-device tasks	202	5	3	0.985	0.976	190	20	0	1.000	0.905	202	8	0	1.000	0.962	0.995	0.947

Table 4: Different scenes and their tracking results. Status means the phone's status in a specific scene (T: On a Table-like Object, H: In Hand). S, F, W stand for tracking succeeded count, tracking failed count, and wrong tracking count respectively. The precision and recall are summarized in Pre and Rec.

Method	Behavior	Range	Sensor	Dataset	Precision	Recall	F-1 Score	Accuracy
DeepAuth [1]	Daily phone usage	On-body	Acc + Gyro	47 users	99.2%	92.7%	95.8%	-
SecurePickUp [35]	Picking up the phone	On-body	Acc + Gyro	24 users	-	-	-	96.3%
Zhang et al. [63]	Stroking on the touchscreen	On-screen	Screen + Ori	138 users	-	-	-	95.0%
Attribute-based(UMDAA-01) [54]	Facing the phone	Around-the-device (≤ 50cm)	Camera	50 users	-	-	-	70.0%
UMDAA-02 Face [39]	Facing the phone	Around-the-device (≤ 50cm)	Camera	48 users	-	-	82.83%	76.15%
PanoTrack	Free movement	Around-the-device (≤ 2m)	Camera	14 users	99.5%	94.7%	97.0%	94.3%

Table 5: An comparison of PanoTrack and representative implicit authentication methods.

Although such a comparison was not well-established because of the differences on sensing modalities and datasets, PanoTrack achieved better or comparative results with existing methods. This validates the effectiveness of our authentication models and our algorithm.

9 STUDY 3: USABILITY EVALUATION

We further evaluated the usability of PanoTrack through a third user study, focusing on the efficiency improvement and user experience.

9.1 Design

The study contained two parts. In the first part, we used a within-subject design to compare the authentication efficiency of Auth+Track (PanoTrack with the mixed strategy) and that of the traditional face authentication method. The independent variable was whether the PanoTrack was turned on (the *on* vs. *off* session). And the dependent variable was the phone access time, *i.e.*, from the moment a user grips the phone to the moment they access the phone content. In each session, participants were randomly assigned a task sequence to simulate a real-life usage experience. There were two types of tasks: 1) phone-usage tasks, including checking the message, opening a browser, making a call, and playing music, and 2) non-phone-usage tasks, including typing on a laptop, stretching themselves, talking with someone nearby (the experimenter), and walking around. The task sequence consisted of these two types alternatively. Each time a task was randomly sampled from the corresponding task set. Participants were asked to perform these tasks until the time of this session reached 10 minutes. The order of the two sessions was counterbalanced. We designed a questionnaire to evaluate each session, which contained questions of NASA TLX [24] and one additional question about their willingness to use in daily life.

In the second part, we conducted a semi-structured interview with each participant to compare Auth+Track (mixed strategy) with SmartLock. After participants experienced Auth+Track in the first part, we further introduced and presented the SmartLock technique, including all three authentication methods: on-body detection, trusted places, trusted devices. We asked participants to try these three methods and ensured that they fully understood how it worked. Then, our interview started with "How do you think about the Auth+Track/SmartLock?" and "Which one do you prefer? Can you elaborate on the reasons?". The experimenter followed up with deeper questions according to participants' responses.

9.2 Participants and Apparatus

We invited the same participants in Study 1 for usability evaluation. The apparatus were the same as Study 1, with SmartLock [20] installed on the phone. In this study, we set the screen locking time as 15 seconds, and face authentication as the default method. Although all participants were familiar with the standard face authentication, none of them had previous experience with SmartLock.

9.3 Procedure

Participants went through a brief introduction and then signed the consent form. In the first part, participants started with one session (either *on* or *off* session), during which their phone access time was recorded. The first session was followed by a short break. Then, participants completed the other session. Each session lasted for 10 minutes. In the second part, After the experimenter ensured that participants understood both Auth+Track and SmartLock, the semi-structured interview was conducted. Finally, participants were thanked and dismissed. The whole study took about 25 minutes and participants received $15 for compensation.

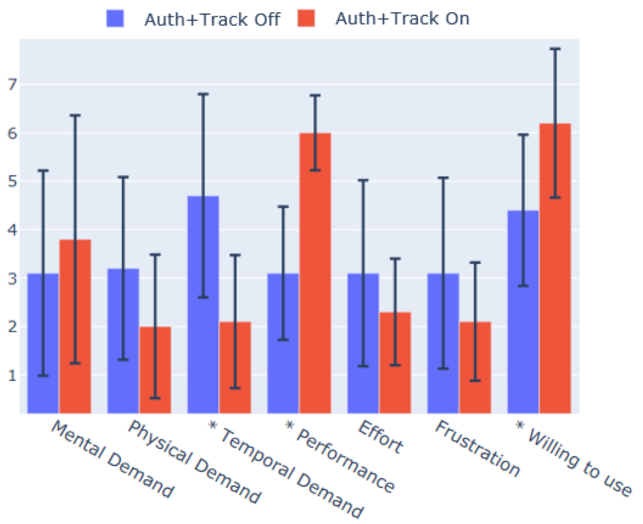


Figure 7: Subjective Ratings of the first part of Study 3. Lower Mental/Physical/Temporal Demand and Frustration scores, and higher Performance and Willing to user scores means better user experience.

9.4 Result

9.4.1 Efficiency. During the *on* session, only 1 participant (P8) lost tracking one time due to cloth obstruction. The Other 9 participants authenticated once and finished all the tasks with quick access, and still were robustly tracked by PanoTrack till the end of the session. The average phone access time in the *on* session (with PanoTrack enabled) was 1.45 seconds (SD=0.47). In contrast, the average access time in the *off* session was 2.98 seconds because of the repeated authentication (SD=0.69). A paired samples t-test showed that PanoTrack was significantly faster than the traditional face authentication method (p < 0.001). Although the task sequence was an accelerated simulation of daily smartphone usage, the results demonstrate how PanoTrack helps users eliminate "gap" authentications and improves their phone access speed.

9.4.2 Usability. The questionnaire results of the experiment present positive feedback from users (see Figure 7). We ran a Wilcoxon signed-rank test for each question. For Auth+Track, users rated significantly lower temporal demand (2.1 vs. 4.7, $p < 0.05$), higher performance (6.0 vs. 3.1, $p < 0.01$), and higher willing to use (6.2 vs. 4.4, $p < 0.05$). Other results did not show significance between the two methods in terms of mental load ($p = 0.59$), physical load ($p = 0.21$), effort ($p = 0.07$), or frustration ($p = 0.12$). The results show that participants acknowledged the advantage of accelerating the authentication process introduced by Auth+Track.

9.4.3 Interview: Auth+Track vs. SmartLock. 8 participants preferred Auth+Track and 2 participants did not have a preference. We summarized the interview results from the aspects of mental load and reliability. 1) Mental load. Participants found SmartLock hard to understand, especially the concept of the trusted location. They were confused how the trusted regions are determined and why this could be secure. Moreover, participants said thinking whether a situation was covered by a trusted device/place was demanding. This was also supported by [44]. *"It's a bother to think whether it's a trusted place when you reach a new place."* (P4). Comparatively, the functionality and the principal of Auth+Track were straightforward easy for participants to understand and it did not require any prerequisites or prior knowledge. 2) Reliability. Participants did not think third-party information required SmartLock (trusted devices) was reliable. The lack of feedback on the on-body detection also worried participants about its reliability. In contrast, participants found the user-centered criterion reliable. The interview results showed that participants accepted Auth+Track as a secure authentication model that is more trustworthy than SmartLock.

10 DISCUSSION AND LIMITATIONS

In this section, we discuss potential solutions and issues related to the practical adoption and deployment of Auth+Track and PanoTrack.

10.1 Possible Solutions to Reduce Mistracking

There are few potential solutions for us to further improve the performance of PanoTrack that worh exploring in the fiture. First, since PanoTrack cannot distinguish whether it is tracking the wrong user, keeping a low mistracking rate is essential for the system. In the user identity assignment evaluation session, mis-tracking happened once (0.5%) because two people were evenly close to and facing the phone. This was hard to be distinguished using a position-based algorithm. However, it could be easily solved by integrating the location of the authenticated face detected by existing face authentication methods. In our current implementation of PanoTrack, we could not access the phone's Face ID system detection results due to the hardware limitation. But we envision this problem could be resolved as these embedded system are becoming more and more accessible. Second, in study 1, we discovered mistrackings were mainly caused by 1) body obstruction from other disturbers, 2) the wrong detection in body keypoints, and 3) low image resolution in far ($> 2m$) distance. A conservative strategy to decrease the mistracking rate in the tracking phase is to restrict the tracking condition to a smaller tracking range (*e.g.*, 2m) or to a stricter criterion dealing with overlapping and obstruction – When a user leaves further than 2m, is obstructed by a disturber, or overlaps with other person in the scene, PanoTrack transits to a "tracking lost" state. If so, the system would yield almost no mistrakcing.

10.2 Form Factor and Energy Consumption

Currently, our implementation is based on an external dual-mode fisheye camera and the whole pipeline runs on a PC server. Our prototype shows the feasibility of Auth+Track, but it is still limited by the hardware size, implementation redundancy, and computational complexity for real-life deployment.

With the development of a low-powered always-on camera on the mobile device (*e.g.*, HUAWEI Mate 30) and mobile processor (*e.g.*, Apple A14 has a NN process unit to accelerate computation), continuous user sensing is becoming a future trend. The implementation of PoseNet using Tensorflow Lite [17], running 60 FPS on mobile devices, also demonstrates the feasibility of user sensing in mobile scenarios. Moreover, various compression techniques in neural networks (*e.g.*, using 8-bit [16] or 16-bit floats [14] in parameter quantization) could further improve the performance and reduce the energy consumption of existing neural inference models for user tracking.

10.3 Privacy Issue

The always-on camera on a smartphone brings about privacy concerns in the following two aspects: 1) privacy of the user and 2) the privacy of other people in the scene. PanoTrack camera directly faces the user's body and the surrounding area, leading to the risk of privacy leak of both the user and surrounding people. There are a few potential solutions.

For 1), As smartphones are getting more and more powerful, an effective approach to alleviate this problem is edge computing [57], *i.e.*, moving the computation from the server to the local phone. Using such a method, the body tracking, hand tracing, and user identity assignment will all take place locally and no data will leave the phone, thus protecting privacy.

For both 1) and 2), the data collecting and image processing procedure should be implemented at a high-privacy level (*e.g.*, at the OS level) or in a customized hardware (*e.g.*, FPGA [3]), so that the always-on camera module outputs semantic information (*e.g.*, keypoints, classifications, heatmap) instead of raw images. (Even the user cannot access the raw data.)

We plan to combine these method to minimize the privacy concerns from users. However, its effect on the tracking performance and accuracy (due to the limited computing power) needs to be investigated in the future.

10.4 System Robustness

Although computer vision algorithms and models show great potential in detecting semantic information in images, they are not 100% reliable yet. Factors including motion blur, improper illumination, interferential scene, and adversarial pattern may lead to detection failure. The robustness of a detection system should be taken into prime consideration before the system is put in practical use. One limitation is that the experiments were conducted in an indoor environment. The performance of PanoTrack algorithms in a more complex environment or against adversarial attacks should be further explored.

To enhance the system robustness and alleviate the bad consequences of detection failure, one solution is to enhance hardware's sensing ability (*e.g.*, using an anti-shake, high-resolution camera) or to enhance model robustness (*e.g.*, training parameters with adversarial noises). Another solution is to sacrifice the algorithm's recall to improve precision. Since there is always a trade-off among multiple factors (hardware investment, computational cost, system precision, *etc.*), finding a Pareto Optimality for practical use is of great importance.

11 CONCLUSION

In this paper, we propose Auth+Track, a novel authentication model that eliminates the "gap" authentication between fragmented smartphone sessions. Auth+Track enables "Authentication Free when User is Around" by sparse authentication and continuous tracking of the user's status. We then present PanoTrack, an instantiation of Auth+Track based on a fisheye camera installed on the top of the phone to capture the panoramic scene, including the user's body and hand. We develop an algorithm pipeline to extract all the key features of the body and the hand for user tracking. The results of our first user study show the good performance of PanoTrack, especially under the around-device scenarios (<2m). Both body tracking and hand statu tracing achieved an accuracy over 99%. In real-life scenarios, PanoTrack achieved an precision of 99.5% and an recall of 94.7%. We compared the PanoTrack against the traditional face authentication method in our second user study. The results of the time measure validated that PanoTrack significantly accelerated the authentication process. Participants also provided positive feedback on PanoTrack. Users accepted "User Around" as a reliable criterion. They believed Auth+Track as a secure authentication model that is trustworthy. We envision Auth+Track has the potential to inspire more authentication-free techniques that lead to human-centered authentication experience.

ACKNOWLEDGMENTS

This work is supported by the Natural Science Foundation of China under Grant No. 61521002, No. 61572276, and National Key R&D Program of China No. 2019AAA0105200, and also by Beijing Key Lab of Networked Multimedia, the Institute for Guo Qiang, Tsinghua University, Institute for Artificial Intelligence, Tsinghua University (THUAI), and Beijing Academy of Artificial Intelligence (BAAI).

REFERENCES

[1] Sara Amini, Vahid Noroozi, Amit Pande, Satyajit Gupte, Philip S. Yu, and Chris Kanich. 2018. DeepAuth: A Framework for Continuous User Re-Authentication in Mobile Apps. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (Torino, Italy) *(CIKM '18)*. Association for Computing Machinery, New York, NY, USA, 2027–2035. https://doi.org/10.1145/3269206.3272034

[2] M. Bicego, A. Lagorio, E. Grosso, and M. Tistarelli. 2006. On the Use of SIFT Features for Face Authentication. In *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*. IEEE, Piscataway, NJ, USA, 35–35. https://doi.org/10.1109/CVPRW.2006.149

[3] Merwan Birem and FranÃğois Berry. 2014. DreamCam: A modular FPGA-based smart camera architecture. *Journal of Systems Architecture* 60, 6 (2014), 519–527. https://doi.org/10.1016/j.sysarc.2014.01.006

[4] G. Bradski. 2000. The OpenCV Library.

[5] Daniel Buschek, Fabian Hartmann, Emanuel von Zezschwitz, Alexander De Luca, and Florian Alt. 2016. SnapApp: Reducing Authentication Overhead with a Time-Constrained Fast Unlock Option. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 3736–3747. https://doi.org/10.1145/2858036.2858164

[6] Z. Cao, T. Simon, S. Wei, and Y. Sheikh. 2017. Realtime Multi-person 2D Pose Estimation Using Part Affinity Fields. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Piscataway, NJ, USA, 1302–1310. https://doi.org/10.1109/CVPR.2017.143

[7] Liwei Chan, Yi-Ling Chen, Chi-Hao Hsieh, Rong-Hao Liang, and Bing-Yu Chen. 2015. CyclopsRing: Enabling Whole-Hand and Context-Aware Interactions Through a Fisheye Ring. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Charlotte, NC, USA) *(UIST '15)*. Association for Computing Machinery, New York, NY, USA, 549–556. https://doi.org/10.1145/2807442.2807450

[8] Liwei Chan, Chi-Hao Hsieh, Yi-Ling Chen, Shuo Yang, Da-Yuan Huang, Rong-Hao Liang, and Bing-Yu Chen. 2015. Cyclops: Wearable and Single-Piece Full-Body Gesture Input Devices. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. ACM, New York, NY, USA, 3001–3009. https://doi.org/10.1145/2702123.2702464

[9] Xiang 'Anthony' Chen, Julia Schwarz, Chris Harrison, Jennifer Mankoff, and Scott E. Hudson. 2014. Air+Touch: Interweaving Touch & In-air Gestures. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. ACM, New York, NY, USA, 519–525. https://doi.org/10.1145/2642918.2647392

[10] Lung-Pan Cheng, Meng Han Lee, Che-Yang Wu, Fang-I Hsiao, Yen-Ting Liu, Hsiang-Sheng Liang, Yi-Ching Chiu, Ming-Sui Lee, and Mike Y. Chen. 2013. IrotateGrasp: Automatic Screen Rotation Based on Grasp of Mobile Devices. In

Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Paris, France) *(CHI '13)*. ACM, New York, NY, USA, 3051–3054. https://doi.org/10.1145/2470654.2481424

[11] Lung-Pan Cheng, Hsiang-Sheng Liang, Che-Yang Wu, and Mike Y. Chen. 2013. iGrasp: Grasp-based Adaptive Keyboard for Mobile Devices. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems* (Paris, France) *(CHI EA '13)*. ACM, New York, NY, USA, 2791–2792. https://doi.org/10.1145/2468356.2479514

[12] Sarah Clinch, Paul Metzger, and Nigel Davies. 2014. Lifelogging for 'Observer' View Memories: An Infrastructure Approach. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication* (Seattle, Washington) *(UbiComp '14 Adjunct)*. ACM, New York, NY, USA, 1397–1404. https://doi.org/10.1145/2638728.2641721

[13] Mauro Conti, Irina Zachia-Zlatea, and Bruno Crispo. 2011. Mind How You Answer Me! Transparently Authenticating the User of a Smartphone When Answering or Placing a Call. In *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security* (Hong Kong, China) *(ASIACCS '11)*. Association for Computing Machinery, New York, NY, USA, 249–259. https://doi.org/10.1145/1966913.1966945

[14] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. 2014. Training deep neural networks with low precision multiplications. arXiv:1412.7024 [cs.LG]

[15] 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* (Seoul, Republic of Korea) *(CHI '15)*. ACM, New York, NY, USA, 1411–1414. https://doi.org/10.1145/2702123.2702141

[16] Tim Dettmers. 2015. 8-Bit Approximations for Parallelism in Deep Learning. arXiv:1511.04561 [cs.NE]

[17] github.com. 2020. PoseNet on TensorFlow Lite. https://github.com/tensorflow/tfjs-models/tree/master/posenet

[18] Mayank Goel, Alex Jansen, Travis Mandel, Shwetak N. Patel, and Jacob O. Wobbrock. 2013. ContextType: Using Hand Posture Information to Improve Mobile Touch Screen Text Entry. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 2795–2798. https://doi.org/10.1145/2470654.2481386

[19] Lorena Gonzalez-Manzano, Jose M. De Fuentes, and Arturo Ribagorda. 2019. Leveraging User-Related Internet of Things for Continuous Authentication: A Survey. *ACM Comput. Surv.* 52, 3, Article 53 (June 2019), 38 pages. https://doi.org/10.1145/3314023

[20] Google. 2014. Google I/O Keynote. https://www.youtube.com/watch?time_continue=1659&v=biSpvXBGpE0.

[21] 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* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 4806–4817. https://doi.org/10.1145/2858036.2858267

[22] Marian Harbach, Alexander De Luca, Nathan Malkin, and Serge Egelman. 2016. Keep on Lockin' in the Free World: A Multi-National Comparison of Smartphone Locking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 4823–4827. https://doi.org/10.1145/2858036.2858273

[23] Marian Harbach, Emanuel von Zezschwitz, Andreas Fichtner, Alexander De Luca, and Matthew Smith. 2014. It's a Hard Lock Life: A Field Study of Smartphone (Un)Locking Behavior and Risk Perception. In *10th Symposium On Usable Privacy and Security (SOUPS 2014)*. USENIX Association, Menlo Park, CA, 213–230. https://www.usenix.org/conference/soups2014/proceedings/presentation/harbach

[24] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In *Human Mental Workload*, Peter A. Hancock and Najmedin Meshkati (Eds.). Advances in Psychology, Vol. 52. North-Holland, Noord-Holland, NLD, 139–183. https://doi.org/10.1016/S0166-4115(08)62386-9

[25] Takahiro Hashizume, Takuya Arizono, and Koji Yatani. 2018. Auth 'N' Scan: Opportunistic Photoplethysmography in Mobile Fingerprint Authentication. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 4, Article 137 (Jan. 2018), 27 pages. https://doi.org/10.1145/3161189

[26] Eiji Hayashi, Sauvik Das, Shahriyar Amini, Jason Hong, and Ian Oakley. 2013. CASA: Context-aware Scalable Authentication. In *Proceedings of the Ninth Symposium on Usable Privacy and Security* (Newcastle, United Kingdom) *(SOUPS '13)*. ACM, New York, NY, USA, Article 3, 10 pages. https://doi.org/10.1145/2501604.2501607

[27] Eiji Hayashi, Oriana Riva, Karin Strauss, A. J. Bernheim Brush, and Stuart Schechter. 2012. Goldilocks and the Two Mobile Devices: Going Beyond All-or-nothing Access to a Device's Applications. In *Proceedings of the Eighth Symposium on Usable Privacy and Security* (Washington, D.C.) *(SOUPS '12)*. ACM, New York, NY, USA, Article 2, 11 pages. https://doi.org/10.1145/2335356.2335359

[28] K. He, X. Zhang, S. Ren, and J. Sun. 2015. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, Piscataway, NJ, USA, 1026–1034. https://doi.org/10.1109/ICCV.2015.123

[29] Ken Hinckley, Seongkook Heo, Michel Pahud, Christian Holz, Hrvoje Benko, Abigail Sellen, Richard Banks, Kenton O'Hara, Gavin Smyth, and William Buxton. 2016. Pre-Touch Sensing for Mobile Interaction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 2869–2881. https://doi.org/10.1145/2858036.2858095

[30] Daniel Hintze, Rainhard D. Findling, Muhammad Muaaz, Sebastian Scholz, and René Mayrhofer. 2014. Diversity in Locked and Unlocked Mobile Device Usage. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication* (Seattle, Washington) *(UbiComp '14 Adjunct)*. ACM, New York, NY, USA, 379–384. https://doi.org/10.1145/2638728.2641697

[31] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, and Hartwig Adam. 2019. Searching for MobileNetV3. arXiv:1905.02244 [cs.CV]

[32] F. Juefei-Xu, C. Bhagavatula, A. Jaech, U. Prasad, and M. Savvides. 2012. Gait-ID on the move: Pace independent human identification using cell phone accelerometer dynamics. In *2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*. IEEE, Piscataway, NJ, USA, 8–15. https://doi.org/10.1109/BTAS.2012.6374552

[33] Ashraf Khalil and Kay Connelly. 2005. Improving Cell Phone Awareness by Using Calendar Information. In *Human-Computer Interaction - INTERACT 2005*, Maria Francesca Costabile and Fabio Paternò (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 588–600. https://doi.org/10.1007/11555261_48

[34] Ajay Kumar and Arun Passi. 2010. Comparison and Combination of Iris Matchers for Reliable Personal Authentication. *Pattern Recogn.* 43, 3 (March 2010), 1016–1026. https://doi.org/10.1016/j.patcog.2009.08.016

[35] Wei-Han Lee, Xiaochen Liu, Yilin Shen, Hongxia Jin, and Ruby B. Lee. 2017. Secure Pick Up: Implicit Authentication When You Start Using the Smartphone. In *Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies* (Indianapolis, Indiana, USA) *(SACMAT '17 Abstracts)*. Association for Computing Machinery, New York, NY, USA, 67–78. https://doi.org/10.1145/3078861.3078870

[36] Fudong Li, Nathan Clarke, Maria Papadaki, and Paul Dowland. 2011. Behaviour profiling for transparent authentication for mobile devices.

[37] Hyunchul Lim, Gwangseok An, Yoonkyong Cho, Kyogu Lee, and Bongwon Suh. 2016. WhichHand: Automatic Recognition of a Smartphone's Position in the Hand Using a Smartwatch. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct* (Florence, Italy) *(MobileHCI '16)*. ACM, New York, NY, USA, 675–681. https://doi.org/10.1145/2957265.2961857

[38] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. In *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (Eds.). Springer International Publishing, Cham, 740–755. https://doi.org/10.1007/978-3-319-10602-1_48

[39] U. Mahbub, S. Sarkar, V. M. Patel, and R. Chellappa. 2016. Active user authentication for smartphones: A challenge data set and benchmark results. In *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. IEEE, Piscataway, NJ, USA, 1–8. https://doi.org/10.1109/BTAS.2016.7791155

[40] Ahmed Mahfouz, Tarek M. Mahmoud, and Ahmed Sharaf Eldin. 2017. A survey on behavioral biometric authentication on smartphones. *Journal of Information Security and Applications* 37 (2017), 28–37. https://doi.org/10.1016/j.jisa.2017.10.002

[41] Ahmed Mahfouz, Ildar Muslukhov, and Konstantin Beznosov. 2016. Android users in the wild: Their authentication and usage behavior. *Pervasive and Mobile Computing* 32 (2016), 50–61. https://doi.org/10.1016/j.pmcj.2016.06.017 Mobile Security, Privacy and Forensics.

[42] Liam M. Mayron. 2015. Behavioral Biometrics for Universal Access and Authentication. In *Universal Access in Human-Computer Interaction. Access to Today's Technologies*, Margherita Antona and Constantine Stephanidis (Eds.). Springer International Publishing, Cham, 330–339.

[43] Liam M. Mayron, Yasser Hausawi, and Gisela Susanne Bahr. 2013. Secure, Usable Biometric Authentication Systems. In *Universal Access in Human-Computer Interaction. Design Methods, Tools, and Interaction Techniques for eInclusion*, Constantine Stephanidis and Margherita Antona (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 195–204.

[44] Masoud Mehrabi Koushki, Borke Obada-Obieh, Jun Ho Huh, and Konstantin Beznosov. 2020. Is Implicit Authentication on Smartphones Really Popular? On Android Users' Perception of "Smart Lock for Android". In *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services* (Oldenburg, Germany) *(MobileHCI '20)*. Association for Computing Machinery, New York, NY, USA, Article 20, 17 pages. https://doi.org/10.1145/3379503.3403544

[45] Matei Negulescu and Joanna McGrenere. 2015. Grip Change As an Information Side Channel for Mobile Touch Interaction. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of

Korea) *(CHI '15)*. ACM, New York, NY, USA, 1519–1522. https://doi.org/10.1145/2702123.2702185

[46] James Nicholson, Lynne Coventry, and Pam Briggs. 2013. Age-related Performance Issues for PIN and Face-based Authentication Systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. ACM, New York, NY, USA, 323–332. https://doi.org/10.1145/2470654.2470701

[47] K. Niinuma, U. Park, and A. K. Jain. 2010. Soft Biometric Traits for Continuous User Authentication. *IEEE Transactions on Information Forensics and Security* 5, 4 (2010), 771–780. https://doi.org/10.1109/TIFS.2010.2075927

[48] Ioannis Papavasileiou, Savanna Smith, Jinbo Bi, and Song Han. 2017. Gait-based Continuous Authentication Using Multimodal Learning. In *Proceedings of the Second IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies* (Philadelphia, Pennsylvania) *(CHASE '17)*. IEEE Press, Piscataway, NJ, USA, 290–291. https://doi.org/10.1109/CHASE.2017.107

[49] Chanho Park and Takefumi Ogawa. 2015. A Study on Grasp Recognition Independent of Users' Situations Using Built-in Sensors of Smartphones. In *Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Daegu, Kyungpook, Republic of Korea) *(UIST '15 Adjunct)*. ACM, New York, NY, USA, 69–70. https://doi.org/10.1145/2815585.2815722

[50] Vishal M Patel, Rama Chellappa, Deepak Chandra, and Brandon Barbello. 2016. Continuous user authentication on mobile devices: Recent progress and remaining challenges. *IEEE Signal Processing Magazine* 33, 4 (2016), 49–61.

[51] V. M. Patel, R. Chellappa, D. Chandra, and B. Barbello. 2016. Continuous User Authentication on Mobile Devices: Recent progress and remaining challenges. *IEEE Signal Processing Magazine* 33, 4 (2016), 49–61. https://doi.org/10.1109/MSP.2016.2555335

[52] Lina Qiu, Alexander De Luca, Ildar Muslukhov, and Konstantin Beznosov. 2019. Towards Understanding the Link Between Age and Smartphone Authentication. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. ACM, New York, NY, USA, Article 163, 10 pages. https://doi.org/10.1145/3290605.3300393

[53] Oriana Riva, Chuan Qin, Karin Strauss, and Dimitrios Lymberopoulos. 2012. Progressive Authentication: Deciding when to Authenticate on Mobile Phones. In *Proceedings of the 21st USENIX Conference on Security Symposium* (Bellevue, WA) *(Security'12)*. USENIX Association, Berkeley, CA, USA, 15–15. http://dl.acm.org/citation.cfm?id=2362793.2362808

[54] 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)*. IEEE, Piscataway, NJ, USA, 1–8. https://doi.org/10.1109/BTAS.2015.7358748

[55] S. Sanderson and J. H. Erbetta. 2000. Authentication for secure environments based on iris scanning technology. In *IEE Colloquium on Visual Biometrics (Ref.No. 2000/018)*. IEEE, Piscataway, NJ, USA, 8/1–8/7. https://doi.org/10.1049/ic:20000468

[56] Toby Sharp, Cem Keskin, Duncan Robertson, Jonathan Taylor, Jamie Shotton, David Kim, Christoph Rhemann, Ido Leichter, Alon Vinnikov, Yichen Wei, Daniel Freedman, Pushmeet Kohli, Eyal Krupka, Andrew Fitzgibbon, and Shahram Izadi. 2015. Accurate, Robust, and Flexible Real-time Hand Tracking. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. ACM, New York, NY, USA, 3633–3642. https://doi.org/10.1145/2702123.2702179

[57] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu. 2016. Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal* 3, 5 (2016), 637–646. https://doi.org/10.1109/JIOT.2016.2579198

[58] Emanuel von Zezschwitz, Alexander De Luca, and Heinrich Hussmann. 2014. Honey, I Shrunk the Keys: Influences of Mobile Devices on Password Composition and Authentication Performance. In *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational* (Helsinki, Finland) *(NordiCHI '14)*. ACM, New York, NY, USA, 461–470. https://doi.org/10.1145/2639189.2639218

[59] Emanuel von Zezschwitz, Paul Dunphy, and Alexander De Luca. 2013. Patterns in the Wild: A Field Study of the Usability of Pattern and Pin-based Authentication on Mobile Devices. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services* (Munich, Germany) *(MobileHCI '13)*. ACM, New York, NY, USA, 261–270. https://doi.org/10.1145/2493190.2493231

[60] Flynn Wolf, Ravi Kuber, and Adam J. Aviv. 2019. "Pretty Close to a Must-Have": Balancing Usability Desire and Security Concern in Biometric Adoption. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. ACM, New York, NY, USA, Article 151, 12 pages. https://doi.org/10.1145/3290605.3300381

[61] Xing-Dong Yang, Khalad Hasan, Neil Bruce, and Pourang Irani. 2013. Surround-See: Enabling Peripheral Vision on Smartphones during Active Use. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 291–300. https://doi.org/10.1145/2501988.2502049

[62] Chun Yu, Xiaoying Wei, Shubh Vachher, Yue Qin, Chen Liang, Yueting Weng, Yizheng Gu, and Yuanchun Shi. 2019. HandSee: Enabling Full Hand Interaction on Smartphone with Front Camera-Based Stereo Vision. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300935

[63] H. Zhang, V. M. Patel, M. Fathy, and R. Chellappa. 2015. Touch Gesture-Based Active User Authentication Using Dictionaries. In *2015 IEEE Winter Conference on Applications of Computer Vision*. IEEE, Piscataway, NJ, USA, 207–214. https://doi.org/10.1109/WACV.2015.35