G2Auth: Secure Mutual Authentication for Drone Delivery Without Special User-Side Hardware

Chuxiong Wu
University of South Carolina
chuxiong@email.sc.edu

Xiaopeng Li
University of South Carolina
xl4@email.sc.edu

Lannan Luo
University of South Carolina
lluo@cse.sc.edu

Qiang Zeng ✉
University of South Carolina
zeng1@cse.sc.edu

ABSTRACT

Because of its cost effectiveness and timeliness, package delivery using unmanned aerial vehicles (UAVs), called *drone delivery*, is drawing growing attention. Authentication is critical for ensuring that a package is not picked up by an attacker's drone or delivered to an attacker. As delivery drones are costly and may carry sensitive or expensive packages, a drone should not get very close to a person unless she is authenticated; thus, conventional authentication approaches that require human-drone physical contact do not work. Existing authentication methods for drone delivery suffer from one or multiple of the following limitations: (1) requiring special user-side hardware; (2) enforcing one-way authentication only; (3) being vulnerable to relay attacks; (4) having compatibility issues. We present the first system, named Greet-to-Auth (G2Auth, for short), that supports mutual authentication between a user and a drone, without these limitations. A user waves her hand holding a smartphone to conduct the authentication. The evaluation shows that it is secure, accurate, usable, and robust.

CCS CONCEPTS

• **Security and privacy → Authentication**; • **Networks → Mobile and wireless security**.

KEYWORDS

Drone delivery, authentication, relay attacks

ACM Reference Format:
Chuxiong Wu, Xiaopeng Li, Lannan Luo, and Qiang Zeng ✉. 2022. G2Auth: Secure Mutual Authentication for Drone Delivery Without Special User-Side Hardware. In *The 20th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '22), June 25–July 1, 2022, Portland, OR, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3498361.3538941

1 INTRODUCTION

The emerging drone delivery service is drawing enormous attention due to its cost effectiveness and convenience. The market is projected to be $29 billion by 2027 [77]. Giant retailers, such as Amazon [3] and Walmart [69], and courier service companies, like UPS [96] and DHL [22], are actively deploying drone delivery. Recently, Amazon obtained FAA approval for drone delivery [13], bringing the technique one step closer to a large number of users. A U.S. startup company, Zipline, has used drones for rapidly delivering life-saving medical supplies, such as blood, in areas with poor infrastructure [1].

Table 1: Comparison. ✓: true, ✗, false, ?: unclear.

Technique	G1	G2	G3	G4
Face/gait/speaker recognition	✓	✗	✗	✓
Google (scanning QR code) [88]	✓	✗	✗	✓
Qualcomm (purchase code) [38]	✓	✗	✗	✓
Walmart (beacon) [67]	✗	✓	✗	✓
SoundUAV [71]	✗	✗	✗	✓
Distance bounding [10]	✗	✓	?	✗
G2Auth	✓	✓	✓	✓

The incoming popularity makes drone delivery an attractive attack target. Among many attacks, *impersonation attacks* are likely against drone delivery [71]. In the case of drone-based courier services, e.g., where a delivery drone collects a package from the sender and delivers it to the designated receiver, attackers can launch at least two kinds of impersonation attacks: (1) *pickup-time impersonation*, where an attacker-controlled drone impersonates the legitimate one in order to steal a package, and (2) *delivery-time impersonation*, where an attacker impersonates the legitimate receiver. This is analogous to real-world impersonation attacks where criminals claim themselves as delivery personnel or legitimate receivers [81]. (3) Moreover, without authentication, it is unclear whether a package (e.g., containing medical supplies or foods) is delivered by a legitimate drone or a malicious one. In order to defeat such attacks, authentication of drones and users is critical.

Delivery drones are expensive and may carry important packages. To prevent an attacker from capturing a drone, it should keep a distance from users until authentication is done, which imposes a unique constraint on authentication. Many conventional authentication approaches that require human-drone physical contact, such as scanning fingerprints, are not secure options.

We aim at the following goals: (G1) **no need of special user-side hardware**; (G2) **mutual user-drone authentication**; (G3) **being resilient to attacks**, such as relay attacks discussed below; and (G4) **no compatibility issues** between drones and user-side

MobiSys '22, June 25–July 1, 2022, Portland, OR, USA

ACM ISBN 978-1-4503-9185-6/22/06…$15.00
https://doi.org/10.1145/3498361.3538941

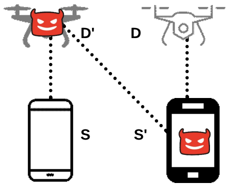


Figure 1: Relay attack. D **(**S**): legitimate drone (smartphone);** D' **(**S'**): malicious drone (smartphone).**

devices. We examine existing authentication approaches that do not require human-drone contact, but none meet all the goals.

We summarize some of the most relevant techniques in Table 1. Face, gait, or speaker recognition can be used for authentication, without involving human-drone physical contact; but there are many known attacks against face recognition [24, 83], gait recognition [35, 36], and speaker recognition [50, 100] (**G3**: ✗). Plus, it cannot authenticate drones (**G2**: ✗) and needs to profile how the user looks/walks/speaks, which harms usability. A Google's patent [88] proposes to authenticate a user by having the drone scan a QR code on the user's smartphone. But it is vulnerable to ***vision relay attacks***, identified by this work: In Figure 1, a malicious drone D', which hovers in front of the legitimate user, can scan the code on her phone S and relay the content to the attacker's phone S'; the latter shows the code to the legitimate drone D. As a result, S thinks D' hovering in front of it is D, and D thinks S' is S (**G3**: ✗). The insecurity of another variant using QR code for drone-delivery authentication is detailed in Section 2.2. In Qualcomm [38]'s patent, a user uses her smartphone to send a one-time purchase code or digital token to the drone. It is vulnerable to *radio relay attacks* (see Section 2.1) [33, 43, 97] (**G3**: ✗). Neither of the two patents considers authentication of drones (**G2**: ✗).

Other approaches require special user-side infrastructure. For example, a Walmart's patent [67] needs the user-side dock/lockbox to be installed (**G1**: ✗). It uses a beacon transmitter and a reader to facilitate authentication, and is vulnerable to radio relay attacks (**G3**: ✗). SoundUAV [71] exploits the fact that the motor noises generated by each drone are unique. A user-side dock installed with a microphone authenticates a drone based on its sound fingerprint. It needs dedicated user-side infrastructure (**G1**: ✗), only supports authentication of drones (**G2**: ✗), and is vulnerable to record-and-replay attacks (**G3**: ✗). It needs per-drone profiling and it is unclear whether the motor noises change over time.

To address relay attacks, distance-bounding protocols [10] are proposed to calculate an upper bound of the distance between participants, based on the fact that radio travels nearly at the speed of light. As the accuracy is sensitive to the slightest processing latency, it requires special hardware [74] that is not widely available yet (**G1**: ✗). Plus, it is unfair and unrealistic to require all people to own high-end hardware. Since its security is still being actively studied [15, 62] and new attacks have been proposed [6] (**G3**: **?**), standard designs and protocols still have a long way to go. Thus, even if a user owns a device that supports a certain distance-bounding protocol, the compatibility issues between a drone and the device cannot be ignored (**G4**: ✗).

Being the first in the literature, we propose a drone-delivery authentication technique, named Greet-to-Auth (G2Auth, for short), that meets all the goals. It does not need special user-side infrastructure but just an ordinary smartphone (**G1**: ✓). A user who holds a smartphone waves her hand to conduct authentication. G2Auth is established on this simple yet solid fact: the IMU (inertial measurement unit) data collected by the user's smartphone during waving, which can be regarded as ground truth assuming the phone is not compromised, and the video data collected by the drone recording the waving operations should correlate, and can be used for mutual authentication (**G2**: ✓). Plus, it is difficult for a *mimicry attacker* (who mimics the legitimate user to wave hand) to closely replicate the waving operations of a legitimate user, as the average human reaction time is greater than 200ms [40, 45, 64], which can be detected as attacks by our system (**G3**: ✓). Furthermore, G2Auth does not cause compatibility issues (**G4**: ✓).

To deliver a accurate, secure and robust solution, the following *challenges* need to be resolved. First, different users wave their smartphones in different ways, causing very different data. It is critical to examine the robustness of the correlation. We thus perform correlation studies about the robustness (Section 3).

Second, it is highly desired that drone delivery can be conducted day and night. When the light level is low (e.g., night time), recognizing a small object and keeping track of it using a camera is still not well resolved in the computer vision area. Moreover, colors of the user's skin/clothes or the background may be similar to that of the held phone. We tried various state-of-the-art object tracking methods, but all failed frequently in such situations. We instead propose a simple yet effective solution to make the system work well in different situations (Section 4).

Third, data from IMUs and cameras are heterogeneous and cannot be compared directly. Based on the object tracking results, we convert the waving trajectory into an acceleration curve. We then propose a series of features and leverage machine learning for correlation calculation (Section 5).

Finally, a determined attacker may practice to mimic a victim user. Defeating such trained mimicry attacks is a challenge. We propose a usable countermeasure by having the user add random pauses when changing the waving direction, effectively defeating trained mimicry attacks.

We build a prototype of G2Auth and perform a comprehensive evaluation (Sections 6 and 7). Below is a subset of the studied questions. Can the system be used to authenticate users *never seen during training*? Can it work at night and in various weather? Is it resilient to mimicry attackers? The evaluation gives positive answers to all the questions. For example, the area under the curve (AUC) is over 0.9988 for users never seen during training, showing a very high accuracy.

This work makes the following contributions.

- We examine existing authentication approaches and illustrate why they cannot be applied to drone delivery. Existing approaches require special user-side hardware, only support one-way authentication, are vulnerable to relay attacks, and/or have compatibility issues. We identify requirements for a drone-delivery authentication system.
- We propose the first authentication approach for drone delivery that meets all the requirements. We resolve multiple challenges to deliver a robust, accurate, and secure design,

which copes with different waving styles, supports authentication during nighttime, compares heterogeneous data, and tackles mimicry attacks.

- We build a prototype system and the evaluation demonstrates its high accuracy, security, robustness, and usability.

The rest of the paper is organized as follows. The system overview is presented in Section 2. A study of the correlation between the IMU data and video data is described in Section 3. Data preprocessing is detailed in Section 4 and the correlation calculation in Section 5. We present data collection in Section 6, the evaluation in Section 7, and the usability study in Section 8. The related work is discussed in Section 9. We discuss the limitations in Section 10 and conclude in Section 11.

2 SYSTEM OVERVIEW

2.1 Background

To prove a legitimate drone is in proximity to a user's phone, an intuitive solution is to use radio characteristics, such as short-range Bluetooth, Received Signal Strength Indicator (RSSI), radio fingerprinting, etc.

However, researchers [32, 33, 43, 97] have repetitively shown the insecurity of these proximity-proving techniques. For instance, the practicality of *radio relay attacks* (aka *Mafia Fraud Attacks* [21]) against the keyless entry system of modern cars, without cracking crypto-keys, has been well demonstrated in the famous work [32]. Car thefts applying relay attacks are not only real [93] but also cheap ($22) [99]. Readers are referred to [19, 72] about the insecurity of applying RSSI, radio fingerprinting, etc.

2.2 Design Choices

Below, we discuss why some more straightforward designs for drone-delivery authentication are not adopted in our system.

Distance Bounding. The concern about the insecurity of the intuitive proximity proving approaches (Section 2.1) has been one of the main motivations of studying distance-bounding protocols [25]. However, as explained in Section 1, they do not meet our goals because of the following issues. (1) It requires special user-side hardware that supports, e.g., UWB (ultra wideband) [39, 51]. It is particularly unrealistic to require all users in rural areas to own high-end devices that support distance bounding. (2) The security of distance bounding is still being actively studied [15, 62], as new attacks are proposed [6]. (3) Due to the lack of standards, the compatibility issue between drones and the diverse user-side devices is a barrier to wide deployment. Instead of relying on distance bounding, our work looks for an inexpensive solution that can be widely deployed without requiring special user-side hardware.

Using QR Code. In order to detect the vision relay attack illustrated in Figure 1 (interpreted in Section 1), one may propose to detect the extra delay incurred by the attack. Specifically, when a QR code is displayed, the user's smartphone S can record the timestamp T_S and the legitimate drone D can record the timestamp T_D when it captures the image containing the QR code. Presumably, the measurement of $T_D - T_S$, when there are no vision relay attacks, should be smaller than the measurement when there are.

The extra delay incurred by a vision relay attack is mainly affected by the malicious drone's camera frame rate and the malicious smartphone's display refresh rate (note the latency due to the extra radio signal relay is 20 μs or less [32], which is negligible compared to the delays discussed below). Assuming the malicious drone uses a camera with fps=240 and a phone with the display refresh rate 144 Hz, the extra latency due to the attack is around 11.1 ms. (A recent study shows that a fast digital camera provides a latency lower than 5 ms and an analog system can make it even shorter [94].) On the side of legitimate users, however, most smartphones today have a display refresh rate of 60 Hz [70], meaning the screen updates one frame every 16.7 ms. After the QR code is shown on the display at its *next* refresh cycle, it is captured by the *next* frame of the drone's camera. As a result, it is difficult to distinguish whether a small extra delay is due to an attack or the display refresh of the user's smartphone and the speed of the legitimate drone's camera.

Blinking Flashlight. Similarly, one may propose to randomly blink the flashlight of the user's smartphone and compare the timestamps recorded by the smartphone and the drone. However, because of the camera latency, even after the flashlight is turned on, it needs the next frame of the camera to record it. Given a low-latency attack system described above, it is difficult to decide whether a small delay is due to an attack. Moreover, an attacker may use a phototransistor, which is used to detect the light, to build an analog system, to make the latency even smaller [32].

The straightforward but insecure designs, such as checking RSSI, using QR code, and blinking the flashlight, illustrate that there are pitfalls for devising an authentication system for drone delivery. The common limitation of checking RSSI, using QR code and blinking the flashlight is that the *element* for authentication can be easily and precisely "cloned" by an attacker. Therefore, it is critical to design an authentication element that can be easily captured by the legitimate entities but difficult to clone.

2.3 Threat Model

Mounting radio relay attacks. Like breaking the keyless entry system of a car [32, 99], relay attacks can fool authentication systems proposed for drone delivery, such as [38, 67, 80] described in Section 1. For example, given a key-protected Bluetooth channel, *without knowing the key*, as illustrated in Figure 1, D' and S' can simply relay the Bluetooth signals between D and S, such that even when D and S are far away from each other, both D and S can be fooled to believe the proximity and conduct the authentication. Our threat model assumes attackers have the capability to launch relay attacks, such that an attacker can use a malicious hovering drone to fool a victim user to start the authentication procedure and relay the encrypted traffic.

There are a variety of opportunities that allow attackers to mount radio relay attacks. (1) It is not uncommon that at a popular place (e.g., a square or apartment building) multiple users wait for their packages (or send out packages). As GPS has inaccuracy near high buildings and bridges [41], this can be profiled and exploited by an attacker. When an attacker notices a delivery drone is approaching, he controls a malicious drone to fool the victim user and meanwhile the attacker impersonates the victim user to fool the legitimate drone. (2) If a routine (e.g., picking up packages around 2pm everyday in a neighborhood) is known by the attacker, he can use GPS spoofing to mislead the drone and send a malicious drone to pick up the packages. GPS spoofing has been a main threat to civilian UAVs [49, 87]. Civilian GPS signals are not encrypted; in GPS spoofing, the attacker transmits fabricated GPS signals with stronger power than the authentic ones, causing the victim receiver to lock onto the attacker's signals [84]. It has been demonstrated on drones [49, 87], and GPS spoofers can be made from inexpensive commercial off-the-shelf components [102]. Recent research shows the difficulty in handling GPS spoofing [84]. (3) When Bluetooth beacons are used for navigation, beacon spoofing [46] can be used to clone the beacons to mislead the legitimate drone.

Mimicry attacks. With the radio relay attack, an *adaptive* attacker who knows how G2Auth works can mimic the user's hand waving in order to fool G2Auth, which we call *mimicry attacks*. As the average human reaction time is larger than 200ms [40, 45, 64] and it is difficult to keep the reaction time consistent, it is not difficult for G2Auth to detect such attacks. An attacker familiar with the target user can practice well to mimic the user better. We call such attacks as *trained mimicry attacks* and study them (Section 7).

Attacks out of scope. The attacker may use a camera to record the user U's waving operations and perform computer vision analysis. The analysis results are then fed into a robot to mimic U, which we call ***robotic mimicry attacks***. The mimicking involves reaction time, due to video analysis, data transmission, planning, and controlling actuators. According to our survey of state-of-the-art robotic techniques, robotic imitation of human actions is actively studied and still very limited. For example, *NAO*, one of the leading humanoid robots, is frequently used by researchers for imitation; despite its high price ($9,000 [78]), it has a delay of 200ms to execute a prescribed motion [31]. Another study shows the end-to-end delay from human-waving to robot-waving is 1.72 seconds [12], much larger than human-to-human imitation. The large reaction time probably cannot be resolved in the near future. We thus do not consider robotic mimicry attacks as a realistic threat.

The attacker may use a camera to record the user's waving operations and play the live video on a screen, which is used to fool D, called *screen-based attacks*. How to distinguish a live person from one shown on a screen is a well-studied question, and there are many software-based anti-spoofing solutions [58, 98, 103] and hardware-based solutions, such as using depth, muti-spectral, or thermal cameras [104]. For example, our experiments find using a cheap thermal camera (HTI-301) can easily distinguish whether the waving hand belongs to a live person or a screen, since the screen does not generate infrared radiation like live persons. Our work assumes one of the existing anti-spoofing solutions is used by delivery drones.

Radio jamming [66] can be used to launch denial-of-service (DoS) attacks. Handling DoS is beyond the scope of this work.

2.4 Main Idea and Assumptions

Main idea. The constraint of no human-drone contact and the threat of relay attacks impose challenges on authentication for drone delivery. We propose an approach that does not require human-drone contact and is resilient to relay attacks. Instead of deploying special hardware to impede attacks, our approach can be used by any users who have smartphones. Specifically, a user holding her smartphone waves her hand a few times to conduct authentication. When the user waves, the IMU of her smartphone generates data, and the camera on the drone records video data. It is evident that the two kinds of data should correlate. Then, information that represents the waving operations is extracted from the two sides, and sent to each other via a key-protected communication channel. Finally, the two sides conduct comparison independently to perform mutual authentication. (Alternatively, assuming the delivery company's cloud server can be trusted, the computation can be offloaded to the server and the result is sent to the smartphone and the drone.)

Assumptions. We assume that the drone D assigned by the courier company and the legitimate smartphone S can establish a key-protected communication channel.[1] There are multiple easy ways for the purpose. (1) Assuming the user has placed a delivery order securely on the courier company's TLS-protected website, the courier company's server generates a key and distributes it to both D and S. (2) The server can send the digital certificate of D to S; then, S and D negotiate a key upon handshaking. (3) The server can be used to bridge the communication between S and D.

We assume D has a camera, a GPS or Bluetooth beacon receiver for navigation, and a wireless network adapter. We assume that, when the drone hovers for authentication, it is easy for a user to identify its camera (e.g., many cameras have a circle of LED lights around them) and stand in front of it. We assume S is installed with the courier company's app.

Authentication procedure. We consider the following representative procedure, although the details may vary depending on the concrete deployment.

(1) The user U places a drone-delivery order using the app installed on the user's smartphone S. After the drone D arrives at the designated location, it hovers and establishes a key-protected communication channel with S. Then, D and S run a clock synchronization protocol [37].

(2) U walks towards the designated location (like using Uber) and unlocks S to confirm the notification.

(3) Facing the hovering drone's camera, U holds S and waves her hand. S conducts IMU data based gesture recognition [52] to recognize waving. Once recognized, S notifies D to start video recording; besides, S generates a short vibration to inform the user of the start of collecting IMU data for authentication. After S collects data of N waving operations (N is studied as a parameter), it generates a long vibration to inform U of the completion of waving and also notifies D.

(4) The captured IMU/video data is exchanged and comparison is conducted independently. If it is a success, the package delivery proceeds; otherwise, it goes back to the previous step until the maximum number of attempts is reached.

It is worth highlighting that, given a drone-delivery task that involves the target user's smartphone S and the designated drone D,

[1] A key-protected channel s assumed in prior existing authentication approaches for drone delivery [10, 38, 67]. Due to radio relay attacks (Section 2.1), a key-protected channel alone is insufficient for authentication.

the authentication compares the data recorded by S and D, meaning that it is a 1-to-1 verification problem, not a 1-to-n identification problem. Its accuracy does not degrade as the user base grows.

2.5 Multiple Drones and Persons

Multiple drones. If a malicious drone D' (or just another delivery drone) hovers near D, it is difficult for the user to decide which is the correct one. Note that even if distance bounding [10] is used, the same issue can arise. Nevertheless, trivial countermeasures can be used to defeat/reveal the attacks. For example, assuming that the multiple drones are not hovering in a vertical line (if yes, the legitimate drone can make a horizontal move slightly; other drones, if they closely follow it, are malicious), D can then generate a notification asking the user to stand right under one of the drones. D then notify the user whether she is under the legitimate drone.

Multiple persons and light sources. There may be people waving hands in the background. As detailed in Section 4, we use a simple but robust method to discard waving in the background: only waving that spans over a threshold portion of the drone's view is considered. Note that even if light sources in the background (e.g., a light swinging due to wind) is considered for comparison, it still needs to pass the correlation calculation (Section 5). In the rare case where multiple persons dispute over a delivery drone, again the trivial countermeasure described above can be used.

3 CORRELATION STUDY

IMU data collected by a smartphone and video data collected by a drone are heterogeneous. How to compare the two kinds of data for computing the correlation score is a question. Second, different people may wave in different ways. Is the correlation computation approach robust? This section answers the two questions.

3.1 Comparing Heterogeneous Data

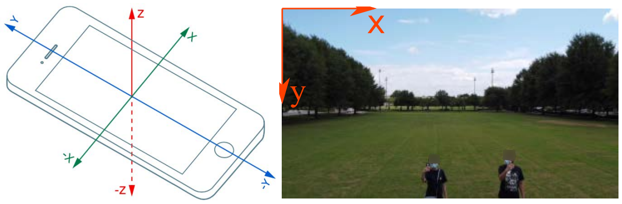
When a user waves, the held smartphone's IMU generates a sequence of acceleration and gyroscope data, and the drone's camera records the trajectory of the smartphone (note that a video contains multiple *frames per second*; e.g., fps = 60).

Failed attempt. To compare the two kinds of data, we first considered this approach: inferring the waving trajectory from IMU data and then comparing it with the trajectory recorded by the video. But fine-grained trajectory inference based on inertial sensor data from smartphones is still an open question [85, 86, 101, 105], as gravity has an impact on the accuracy of orientation projection and double integration of the acceleration gets worse over time.

Comparing acceleration. Our observation is that based on video frames, acceleration can be calculated from the smartphone's displacement, as the former is the second derivative of the latter. On the side of smartphone, its IMU directly generates acceleration data. Thus, the acceleration data can be the basis of comparison. But the acceleration data still cannot be compared directly for two reasons.

First, the units are different, as the unit for acceleration on the smartphone side is m/s^2, while that on the drone side is $pixel/s^2$. To resolve it, we normalize the data between −1 and 1, such that they can be compared in a uniform scale.

Second, as illustrated in Figure 2, the two sides use different Cartesian coordinate systems. The coordinate system of the accelerator in a smartphone is relative to the smartphone itself, which means that when the smartphone is waved, the three axes may change relatively to the earth. On the drone's side, it hovers in front of a user to record waving operations; we define the axis along the width of a video frame as the x-axis and the one along the height as the y-axis. If a smartphone is held vertically and right in front of the drone, the two coordinate systems align well. However, it is not realistic to expect all users wave phones that way. Thus, it is important to examine whether data correlation exists, regardless of how a phone is waved. To that end, in Section 3.2, we perform empirical studies to examine the robustness of correlation.



(a) Smartphone (b) Drone's camera

Figure 2: Different coordinate systems.

3.2 Robustness of Correlation

We first assume that a user holds her smartphone vertically and waves her smartphone horizontally (we will show that this assumption is not necessary); our observation is that, when the user changes the waving direction (e.g., from left to right), the IMU-collected acceleration value along the x-axis reaches either its peak or valley.[2] More generally, we hypothesize that, regardless of the posture of the held smartphone and the waving trajectory, along at least one of the three axes in the smartphone's coordinate system, the IMU-collected acceleration value will reach its peak or valley as the waving direction changes, since it is unlikely that the accelerometer does not sense the direction change along any axis.

To verify this hypothesis, we design an empirical study. We decompose an waving operation into two aspects: (1) Holding posture: how a user holds her phone in hand; we consider three postures in the study: vertical, diagonal, and horizontal, as shown in the three photos on the left of Figure 3; (2) Waving direction: how a user waves her phone; we consider three directions: "left-right", "diagonal", and "up-down". Participants then enumerate all combinations of the two aspects (totalling nine) to wave smartphones. Note that during testing of our system, users are not limited to the nine waving styles, as long as the user does not wave the phone "forward-backward", since "forward-backward" waving would cause little displacement from the view of the drone's camera.

After data preprocessing (Section 4), for each of the nine combinations, we plot the acceleration data from the two sides (smartphone and drone). E.g., the sub-figure in the upper left corner of Figure 3 is based on the waving operations when a user holds the phone vertically and waves it "left-right" (i.e., waving it horizontally). It shows that the IMU's acceleration data along the x-axis (denoted as a solid red line) correlate well with the acceleration data derived from the video. *This correlation exists consistently across all the nine combinations, although the correlated axes vary.* (Depending on the

[2] We do not make assumptions on how the two coordinate systems define "positive" and "negative", as we use the absolute value of the correlation measurement (Section 5).

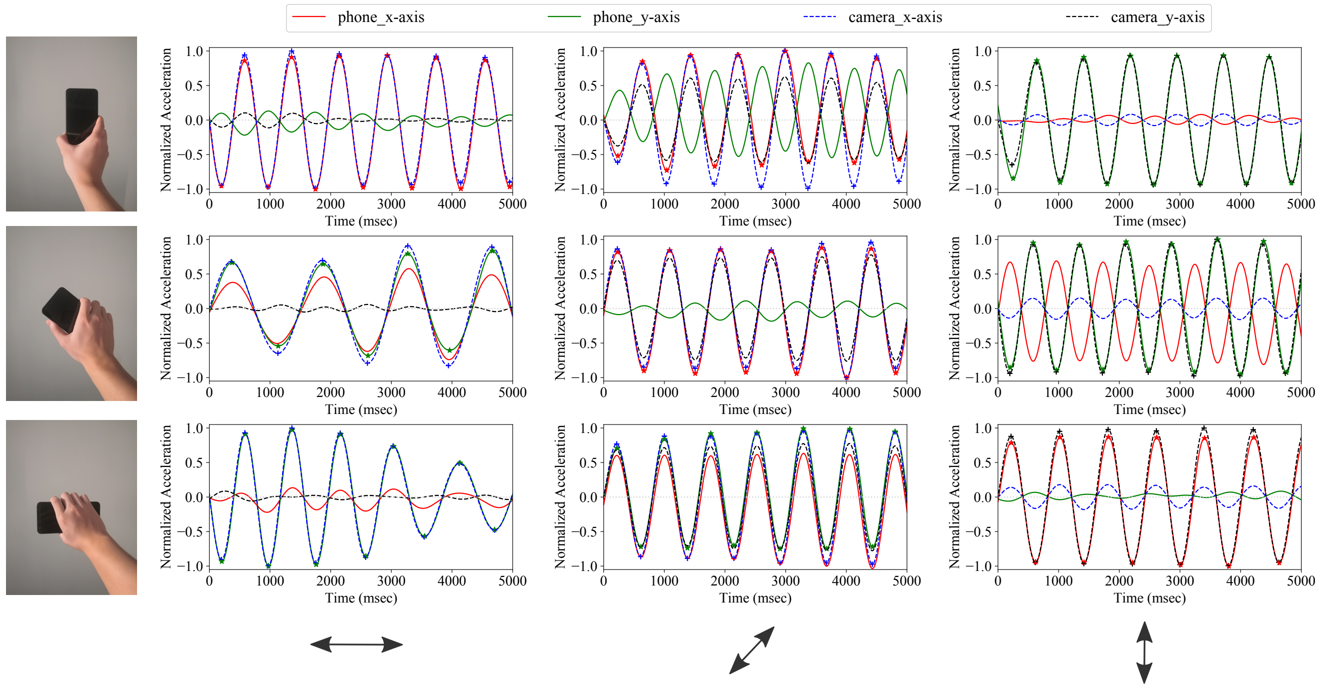


Figure 3: One of the correlation studies. It examines three holding postures: vertical, diagonal, and horizontal, and three waving directions: "left-right", "diagonal", and "up-down". They lead to nine (9) combinations (each illustrated in a sub-figure regarding the acceleration data from the phone and the drone), showing high correlation between acceleration from the two sides.

waving operations, the IMU-collected acceleration data along *z*-axis may also correlate with the direction changes well; to make the illustrations clear, we did not include the data.)

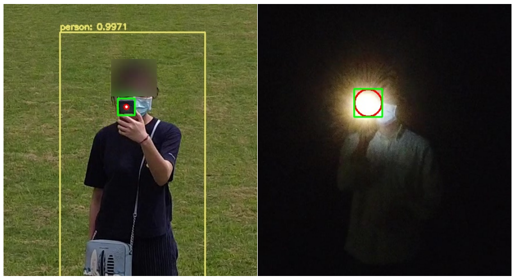
We then vary the waving trajectory by using an "arc" motion (i.e., elbows or shoulders as the points of rotation), the waving speed, and involve 20 participants of different genders and ages ranging from 18 to 67; the correlation always exists along at least one axis. The axis whose data shows the largest peak-valley changes during waving is called the *primary axis*. This applies to both the IMU and video-derived acceleration data. Based on the empirical studies, we conclude that *the IMU-collected acceleration data along the primary axis correlates well with the video-derived acceleration data along its primary axis.*

4 DATA PREPROCESSING

We discuss how to extract acceleration and preprocess data. The output is two sequences of normalized acceleration data, one from IMU and the other from a video.

4.1 Obtaining Trajectory from Video

To obtain the waving trajectory from the video, we first tried to detect a smartphone and employ object tracking to track its movement. But when a drone hovers with a secure distance (e.g., ≥5m) away from users (Section 6), making the smartphone a small object. While both object detection and object tracking are actively studied and many solutions have been proposed [16, 17, 53], accurately detecting and tracking small objects are still open questions [26].



(a) Daytime (b) Nighttime

Figure 4: Images taken by a drone's camera (the images are cropped from large-sized video frames).

We propose a two-step solution: (1) Our system first performs person detection, which can be made very accurate. In this step, we use *YOLO_V3*, one of the fastest and most accurate object detection algorithms [75]. (2) G2Auth's mobile app installed on the user's smartphone automatically keeps the flashlight on during waving. (G2Auth requires the smartphone's back to face the drone during waving.) As shown in Figure 4(a), within the bounding box for the detected hand-waving person, our system searches for a small bright area (using contour detection [5]) to locate the possible positions of the flashlight; this step may locate multiple small bright areas, which usually do not show the waving movements and thus can be excluded easily.

During the daytime, person identification in Step (1) is necessary as there may exist many small bright areas (like cloud, metal and

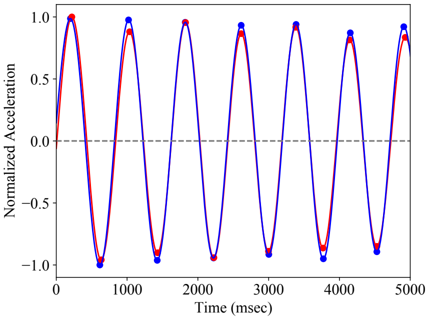


Figure 5: Preprocessed data (red: phone; blue: drone).

glass), and we can use the bounding box output by person detection to narrow down our search. During the nighttime, person detection is not needed, since the flashlight distinguishes itself from the surroundings, as shown by Figure 4(b).

Once the flashlight is located, we start object tracking with a square bounding box covering the flashlight, as shown in Figure 4. It is interesting to note that we are *not* tracking the smartphone, as a phone may be partially covered by the user's hand and have a color similar to the user's clothes or background, leading to tracking failures (Section 7.1). Instead, we are tracking a flashlight, which has salient features. We apply a state-of-the-art small object tracker, PrDiMP [18], which achieves the best performance in the UAV123 dataset [18], containing many small and fast moving objects. The output of object tracking is the trajectory of the smartphone. Finally, only waving that spans over a threshold portion P of the view width is considered (our empirical study finds $P = 1/15$ works well). The purpose is to prevent G2Auth from considering waving in the background and other light sources in the vicinity of the user. Note that even if *huge* waving light sources in the background (e.g., a light swinging due to wind) is considered for comparison, it still needs to pass the correlation calculation (Section 5).

In short, while small object detection and tracking are still challenging problems in general, we exploit the uniqueness of our authentication procedure (i.e., waving) and the hardware capability (i.e., flashlight) to deliver a robust solution.

4.2 Preprocessing Trajectory and IMU Data

The trajectory data output by object tracking may fluctuate and contain noises, and so do the acceleration data collected by IMU. We thus perform the following preprocessing: (1) Linear interpolation: gaps in the data due to uneven sampling can be filled. (2) A Low-pass Butterworth filter [82] with a cut-off frequency of 3Hz is used to filter out noises. The frequency of waving is less than 3Hz, so this does not harm critical information about waving; the noise caused by vibrations of human body has a frequency greater than 3Hz [76] and can be removed. (3) After the trajectory is preprocessed, we get the acceleration value at any moment by computing the second derivative of displacement. (4) Given the two sequences of acceleration data that have different physical units (m/s^2 from IMU and $pixel/s^2$ from video), to make them comparable, we normalize the data of each sequence in the range of $[-1, 1]$, as shown in Figure 5.

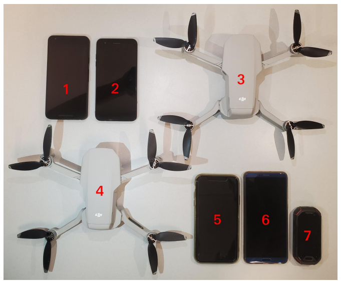


Figure 6: Seven devices used in our experiments.

5 CORRELATION CALCULATION

After getting the two sequences of normalized acceleration data, we check whether the two sequences correlate with each other, in order to determine whether the authentication is a success or not.

We consider two methods. The first method uses Pearson correlation coefficient (PCC) [8], one of the most widely used algorithms for calculating the correlation of two sequences. We use its absolute value as the *correlation score*, as the two coordinate systems (the smartphone's and drone's) may have opposite definitions about "positive" and "negative." This method then uses thresholding to determine whether the authentication is a success.

The second method is based on machine learning. We use the correlation score as one of the multiple features. To extract features from two sequences of acceleration data, we first define *critical events* as the peaks and valleys in the curve of the acceleration data, and obtain the timestamps of these events, as shown in Figure 5. Our *insight* is that the two sequences of critical events should align well in terms of their occurrence time. E.g., an attacker may happen to "hit" some timestamps of critical events, but the variance of time difference between critical events from the two sides tends to be high. Given the timestamp sequence on the smartphone side $S_P = \{t_P^{(1)}, t_P^{(2)}, \ldots, t_P^{(n)}\}$ and that on the drone side $S_D = \{t_D^{(1)}, t_D^{(2)}, \ldots, t_D^{(n)}\}$, we generate the following features (in addition to the correlation score): (1) *Time difference values*: for each t_P in S_P, we find a t_D in S_D that is closest to t_P, and calculate the difference between t_P and t_D; (2) *Non-correlated event number*: the number of extra t_D in two consecutive timestamps in S_P; (3) *Standard deviation*: standard deviation of the time difference values; (4) *MAD*: median absolute deviation of the time difference values; (5) *Modified z-score*: modified z-score of the time difference values.

Regarding the classifier, we consider three: support vector machine (SVM), k-Nearest Neighbors (kNN), and Random Forest (RF). Our final design chooses the second method and adopts SVM because of its best performance (Section 7.3).

6 DATA COLLECTION

The research was conducted under an IRB approval and followed the CDC guidance about COVID-19 (e.g., wearing masks and using hand sanitizer). To evaluate the system we collected multiple datasets. We recruited 20 participants,[3] whose ages range from 18 to 67, 10 males and 10 females, including undergraduates, graduates, faculty members, janitors, and retired people, in our experiments.

[3] "*Training Dataset Size*" (Section 7.3) shows why they are sufficient.

6.1 Devices

Figure 6 shows the devices used in our experiments, including two DJI Mavic Mini drones and five smartphones. The DJI Mavic Mini drone is positioned as a beginner camera drone. We use the built-in camera of each DJI Mavic Mini drone to capture users' hand movements. The camera resolution of one drone is set to 2.7K at 30 FPS, and that of the other is set to 1080P (1920 × 1080) at 60 FPS. (In Section 7.3, we study the impacts of camera resolution and FPS on the system performance.) We use Nexus 5X and LG K8 to collect data for building *Dataset I* and *Dataset II*. The other smartphones (i.e., iPhone 11, Honor View 10, and Unihertz Atom) are used for the parameter study (Section 7.3).

6.2 Dataset I for Accuracy Evaluation

Experimental Setting. To build *Dataset I*, we use both the Nexus 5X and LG K8 smartphone. We randomly and equally assign the two phones to the participants (i.e., 10 participants uses the Nexus 5X and another 10 the LG K8). We deploy two drones to record each participant's hand motions simultaneously. The drones hover next to each other, and we set their height H to 4 meters.

We ask each participant to stand 5 meters (horizontal distance D) away from the drones, where $H = 4$ and $D = 5$. Each participant holds a smartphone and performs the authentication operations in front of the drones, for 30 times. The participants are allowed to wave the smartphone in a way most comfortable to them using their dominant hand.

Positive Pairs. When a participant performs the authentication operations in front of the drones, we collect one positive data pair *for each drone*: one is the acceleration data from the smartphone, and the other a video captured by this drone. For each drone, we collect 600 (= 20 × 30) positive pairs, each with a label $s = 1$.

Negative Pairs. Assuming two users, μ_1 and μ_2, authenticate to the drones D_1 and D_2, respectively, the accelerometer data S_{P_1} from μ_1's smartphone and the video S_{D_2} captured by D_2 constitute a negative pair; also, the accelerometer data S_{P_2} from μ_2's smartphone and the video S_{D_1} captured by D_1 constitute another negative pair.

To build such an uncorrelated sample, we perform *time alignment* for each pair of authentications, randomly selected from two users, such that the authentications can be considered as starting nearly at the same time. As studies [40, 45, 64] have demonstrated that, even for athletes, the best audio/visual reaction time of human is greater than 50ms (generally between 100–300ms), we shift the timestamps of S_{P_1} to make the starting time difference between S_{P_1} and S_{D_2} within the range of $[-300, -50]$ms or $[50, 300]$ms. The same time alignment is also performed for S_{P_2} and S_{D_1}. We finally generate 600 negative pairs, each with a label $s = 0$.

6.3 Dataset II for Security Evaluation

Experimental Setting. To build *Dataset II*, we divide the 20 participants in *Dataset I* into two parts: 10 act as victims and the others 10 as attackers; one victim and one attacker form a pair. Thus, there are 10 pairs of victims and attackers. We consider two types of *mimicry attacks* (**MA**), **MA-untrained** and **MA-trained**, as discussed in *Threat Model* (Section 2.3).

We provide the attacker $\mathcal{A}$ with a *clear view* of the victim $\mathcal{V}$'s hand movements, by letting $\mathcal{A}$ stand next to $\mathcal{V}$ (1 meter away). We use the same drone to capture their waving operations together. The camera resolution of the drone is set to 2.7K at 30FPS, and its height is set to 4 meters. $\mathcal{A}$ and $\mathcal{V}$ are 5 meters (horizontal distance) away from the drone ($D = 5$).

MA-untrained. We tell them the purpose of this experiment: an attacker mimics the victim's hand movements to fool our system, and explain how our system works. We ask the victim to perform authentication operations and the attacker to launch the mimicry attack simultaneously. The attack is repeated for 15 times, with one pause introduced in each authentication procedure, and another 15 times without pauses. Here, a *pause* means the user pauses the waving for a *random* short time intentionally prior to changing the waving direction. Our experiment data show that 700ms works well as a threshold for detecting a pause as an intentional pause.

For each authentication, we construct a data pair consisting of S_{P_V} and S_{D_A}, where S_{P_V} is the acceleration data from $\mathcal{V}$'s smartphone and S_{D_A} is the video captured for $\mathcal{A}$'s hand movements. We collect 150 (= 10 × 15) pairs for the authentication operations without pauses, and the same number of pairs for the authentication operations with pauses.

MA-trained. We first ask each victim to perform authentication in front of the drone for 5 times, and record a video of each authentication. Each attacker is trained by watching videos as many times as needed. The attacker only needs to learn one victim's actions and launch attacks against that victim. During training, we provide the attackers with feedback on the differences between their hand movements and the victims', so that they can adapt their operations.

After the attacker feels confident enough, the victim performs authentication operations and the attacker launches the mimicry attack simultaneously. Their hand movements are recorded by the drone's camera, at the same time. Similar to ***MA-untrained***, each pair of attacker and victim performs the authentication operations with and without pauses for 15 times. We collect 150 (= 10×15) pairs for the authentication operations without pauses, and the same number of pairs for the authentication operations with pauses.

7 EVALUATION

Section 7.1 presents the accuracy of G2Auth, and Section 7.2 studies its resilience to mimicry attacks. Section 7.3 presents a detailed parameter study, and Section 7.4 the efficiency.

Metrics. We use False Rejection Rate (FRR) and False Acceptance Rate (FAR) to evaluate the performance of G2Auth. A lower FRR indicates that the system makes fewer mistakes for authorized users, resulting in better *usability*. On the other hand, a lower FAR indicates better effectiveness of the system in preventing adversaries from gaining access. We also report Equal Error Rate (EER) and Area Under the Curve (AUC) of Receiver Operating Characteristics (ROC) curve: EER reports FRR (FAR) when FRR=FAR, while AUC provides an aggregate measure of performance across all possible thresholds [90].

7.1 Authentication Accuracy

We use *Dataset I* to test the accuracy of G2Auth. Similar to many previous works on evaluating authentication systems [29, 55], we adopt a strict mechanism, Leave-One-Subject-Out (LOSO), to obtain the average performance over all subjects. In LOSO, we iteratively choose one subject for testing and use the data of the other 19

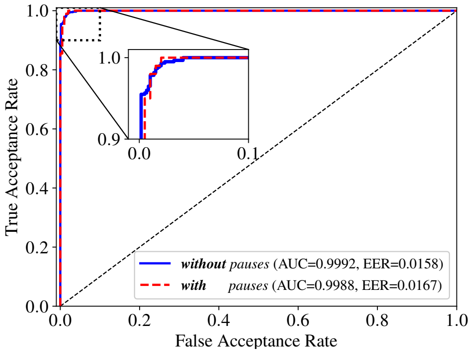


Figure 7: ROC curves, AUC, and EER.

subjects to train the system. We compute the average performance over all the subjects. Through this, we can examine whether our system is *user agnostic*—whether it can work for users never seen during training. We present the results using the data for the drone whose resolution is set to 2.7K at 30 FPS (the impact of ***Camera Resolution*** is studied in Section 7.3). We choose the number of waving operations as 8 (also studied in Section 7.3).

In our system design, if the tracking algorithm fails to track the phone, we ask the user to re-authenticate, instead of moving forward to the next step of data correlation calculation. Thus, failure of tracking has an impact on the system's FRR, but no impact on the system's FAR. We define FRR_{sys} as the system's FRR, which consists of two parts: FRR_t and FRR_c. FRR_t is the failure rate of the tracking algorithm, and FRR_c is the FRR of the system excluding the tracking algorithm. We thus have: $FRR_{sys} = 1 - (1 - FRR_t) \times (1 - FRR_c)$.

When the flashlight is turned on, the tracking algorithm we adopted, PrDiMP [18], can achieve a success rate 0.98 ($FRR_t = 0.02$) for the daytime and a success rate 1.0 ($FRR_t = 0$) for the nighttime. In contrast, when the flashlight is turned off, $FRR_t = 0.20$ for the daytime and $FRR_t = 0.74$ for the nighttime. In the following presentation we refer to FRR_c as FRR, unless otherwise stated.

To evaluate the system performance on data with pauses, we apply the model trained using *Dataset-I*, which only contains data without pauses, to test the data with pauses. Specifically, we ask 10 participants to perform authentication operations with random pauses for 15 times, and use the model trained using *Dataset I* to test the new collected data.

Figure 7 shows the ROC curves for experiments with and without pauses. Without pauses, our system achieves an average $EER = FRR = FAR = 0.0158$ and $AUC = 0.9992$. The low EER indicates that G2Auth can distinguish authorized accesses from unauthorized ones with a high accuracy (=1-EER) of 0.9842. (When only PCC is used, which is the first method described in Section 5, $EER = 0.0283$.) With pauses, G2Auth achieves $EER = 0.0167$ and $AUC = 0.9988$. Thus, we find the model trained using the data without pauses can be directly applied to testing data with pauses, and the high correlation exists regardless of pauses.

We analyse the very few false rejection cases and find that they are mainly caused by the inaccuracy of tracking. For example, there are cases when the sunlight passing through the leaves form small bright spots that look similar to the flashlight (see **Object Tracking Algorithm** in Section 7.3). In the second attempt, the drone can actively turn around and successfully finish the authentication.

Table 2: Phone tracking success rates.

Algorithms	Flashlight off		Flashlight on	
	day	night	day	night
CSRT [59]	0.58	0.12	0.86	1.00
ECO [17]	0.74	0.22	0.95	1.00
RT-MDNet [47]	0.74	0.24	0.96	1.00
ATOM [16]	0.76	0.24	0.96	1.00
DiMP [9]	0.76	0.26	0.98	1.00
PrDiMP [18]	0.80	0.26	0.98	1.00

Our current prototype uses a simple method for clock synchronization [37]. The resulting clock difference, measured using the method [27], is $1.7ms(\pm 0.9ms)$. This is much shorter than the average human reaction time >200ms [40, 45, 64].

7.2 Resilience to Mimicry Attacks

This section evaluates the resilience of G2Auth (based on the threshold selected in Section 7.1 that achieves EER = 0.0158) to mimicry attacks. We use *Dataset II* in this experiment, where 10 participants act as victims and the other 10 as attackers (see Section 6.3 for details).

Resilience to MA-untrained. Without pauses introduced during authentication, G2Auth can successfully identify 91% (= $1 - FAR = 1 - 0.09$) of the attacks, on average. The performance can be greatly improved if the pauses are added—on average, 98% (= $1 - FAR = 1 - 0.02$) of the attacks can be identified by G2Auth. The results demonstrate that pauses during authentication can increase the difficulty for attackers in mimicking the victims' hand movements. Thus, the authentication operations with pauses are more secure.

Resilience to MA-trained. Under MA-trained attacks, the attackers' success rate increases sharply—G2Auth can only identify 74% (= $1 - FAR = 1 - 0.26$) of attacks on average, which reveals a weakness of authentication without pauses, under trained attacks. To enhance the resilience to MA-trained, G2Auth requires that users to intentionally add *at least* one pause. This is enforced automatically by checking whether the acceleration reaches zero for a short time. All the participants successfully followed the instructions by adding at least one pause in each authentication procedure, which indicates that adding random pauses is not a problem to the users. Then, the attackers' success rate is reduced from 0.26 to 0.04—G2Auth can successfully identify 96% (= $1 - FAR = 1 - 0.04$) of attacks, on average. Thus, the pauses decreases the attackers' success rate by making the waving operations more unpredictable and difficult to mimic.

More Pauses. When collecting data with pauses, users were free to decide the number of pauses (but at least one). We then investigate how the number of pauses affects the attacker's success rate. We find that ***when it increases to three (3), the FAR under MA-trained attacks becomes zero***, while FRR is below 0.019. Thus, to achieve high security, a delivery company can enforce the number of pauses ≥ 3.

7.3 Parameter Study

Object Tracking Algorithm. G2Auth needs to track smartphones through computer vision analysis. There has been much research on object tracking. This experiment evaluates some state-of-the-art algorithms, including CSRT [59], ECO [17], RT-MDNet [47], ATOM [16], DiMP [9], and PrDiMP [18]. To evaluate the tracking success rates, we manually check the bounding box during tracking with 600 videos about phone waving (e.g., whether it instead tracks an object in the background). The result in Table 2 shows that the tracking success rate gets improved greatly for all the algorithms when the flashlight is turned on, especially in nighttime. We choose PrDiMP for its high performance; its failed cases share a common feature—they have bright spots in the background, e. g., mottled sunlight through tree leaves. Therefore, we suggest users avoid authenticating under such background.

Classifier. We train the model with different classifiers, including SVM, kNN and Random Forest. For SVM, we examine the linear, polynomial and radial basis function (RBF) kernels, and finally adopt RBF; after grid search, we set the optimal hyperparameter, c as 20 and γ as 0.01. For kNN, we test different values of k, ranging from 1 to 20, and find 3 the optimal value. For Random Forest, we test different number of trees, ranging from 50 to 200, and select the optimal value as 120. The results (EER_{SVM}=0.016, EER_{RF}=0.018, $\text{EER}_{k\text{NN}}$=0.021) show that SVM has the lowest EER.

Number of Waving Operations. More operations provide better security but also require longer time to authenticate, which harms usability. Figure 8(a) shows the EER with varying number of events. As expected, EER decreases as the number of events increases. We chose eight, considering both security and usability.

Training Dataset Size. We evaluate the impact of training dataset size on the system performance. The training dataset size is defined as ***the number of participants*** for training, denoted as m, whose samples are used for training. We train G2Auth with m ($1 \leq m \leq 19$ with a step of 2) participants' data and test it with the data of the rest of the participants ($20 - m$). Figure 8(b) shows the results. It can be seen that the accuracy of the classifiers converges, given $m \geq 15$.

Camera Resolution. By downsampling the resolution of 2.7K (2720×1530), we get different camera data with a resolution of 1080P (1920 × 1080) and 720P (1280 × 720). We then evaluate the system performance in terms of different camera resolutions. As shown in Figure 8(c), the higher the resolution, the better performance (the lower EER) of G2Auth. Even with a low resolution (i.e., 720P), G2Auth can still achieve a satisfactory accuracy (EER = 0.02).

Camera FPS. To measure the impact of FPS, we use the data captured by the DJI Mavic Mini drone, with the camera resulution set as 1080P at 60 FPS. By downsampling the frame rate of the videos, we get different camera data with different FPS. We then evaluate the performance of G2Auth in terms of different FPS. The results show there is a significant improvement when FPS is increased from 15 to 20, but the EERs improves little when FPS further increases. FPS≥ 20 can be satisfied by most cameras today [57].

IMU Sensor Sampling Rate. A higher sampling rate can capture subtler characteristics of the IMU sensor data, but it also introduces higher burdens (e.g., data collection and communication). To find the optimal sampling rate for the IMU sensor of smartphones, we study the sampling rate, ranging from 10Hz to 100Hz, at a step of 10Hz by downsampling the original sensor data. Figure 8(d) shows the result. We can see that when the sampling rate increases from 10HZ to 20Hz, the performance increases significantly. When the sampling rate is higher than 40Hz, the performance tends to be stable. We thus select a sampling rate of 50HZ, which can be satisfied by most IMU sensors [4, 65].

Smartphones. Besides the two smartphones that were used to collect *Dataset I* and *Dataset II*, we select three more as shown in Figure 6: (1) a very small Android Phone, Unihertz Atom, with 96 × 45 × 18 mm in dimension and 108 grams in weight, (2) a large Android phone, Honor View 10, with 157 × 75 × 7 mm in dimension and 172 grams in weight, and (3) iPhone 11 with 150.9 × 75.7 × 8.3 mm in dimension and 194 grams in weight. No significant difference is observed in the performance between the three smartphones. We can thus conclude that the smartphone size, weight, and operating system have little impact on the performance.

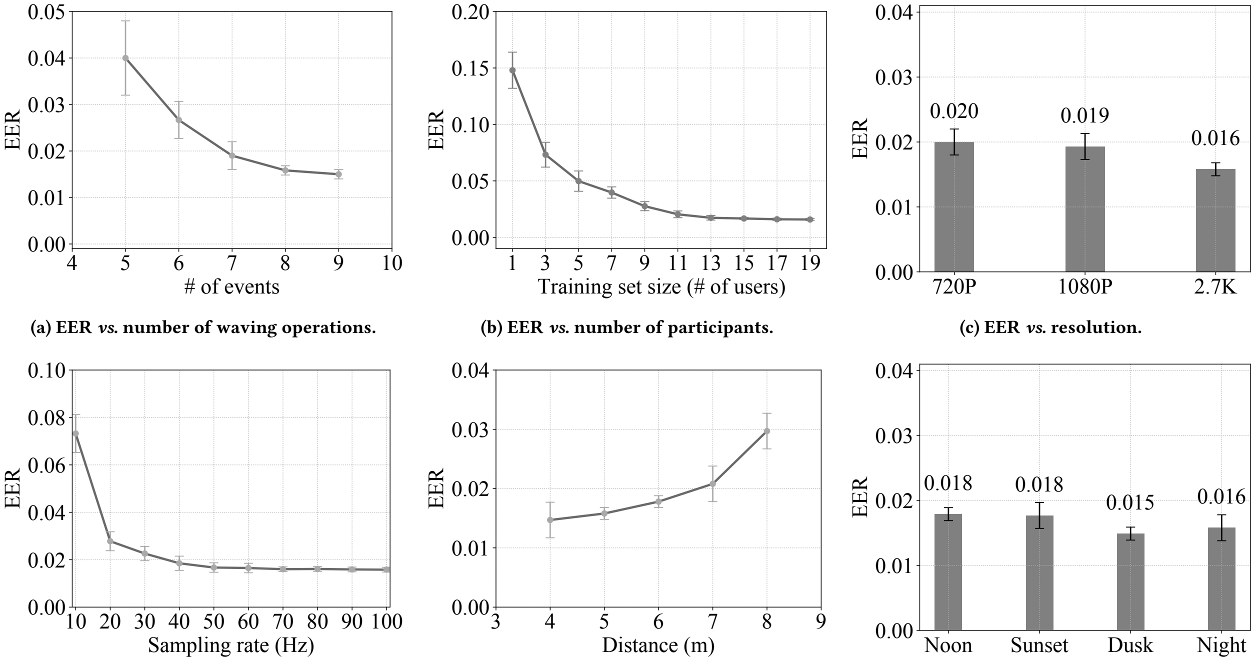
Horizontal Distance Between User and Drone. We test the stability of G2Auth on different horizontal distances between the user and drone. The horizontal distance D is selected from 4 to 8 meters. We invite 10 participants; each performs the authentication operations 15 times for each distance. In Figure 8(e), when D is increased from 4 to 7 meters, no significant difference is observed; the performance decreases greatly when D reaches 8 meters. $D \geq$ 5m is far enough to avoid physical attacks for an attacker captures the drone. We thus select D = 5m.

Illuminance Level. To evaluate the impact of illuminance to the performance of our system, we collect data based on different times of the day: (1) *Noon*, (2) *Sunset*, (3) *Dusk*, and (4) *Night*. Figure 8(f) illustrates the results. G2Auth tends to work slightly better when the light level is low, probably because the flashlight distinguishes itself better in such cases. But no significant difference is observed, showing our system can work under different light levels.

Different Weather. The data collection took multiple weeks, during which there were various weather conditions, such as sunny, cloudy, light rain, and misty. The testing results show that the data collected in different weather have negligible impacts. This is consistent with the overall AUC near 1 (Section 7.1). We attribute it to the advances in flight stabilization and wide deployment of drone gimbals [2, 95], which lead to stable hovering and videos under different weather.

Gender and Age. We group the testing data according to the gender, and finds it has little impact on the accuracy: EER_{male} = 0.0161 vs. $\text{EER}_{\text{female}}$ = 0.0155; so does the age.

View Angle. We assume a delivery drone's camera is easy to identify (see Section 2.4) and during our data collection we find participants are able to stand right (or very close) in front of the camera. Still, we are interested in studying the impact of different angles to the accuracy. A view angle is 0 degree if the user stands right in font of the camera, and it increases as the user stands away from that direct sight. More formally, it is the azimuth angle from the point of view of the camera. The DJI Mavic Mini drone used in our experiments supports a 83 degree field of view (FOV), we collect data by varying the view angle from 0 to 25 degrees with a step of 5 degrees. The results show that the angle of view has a very small impact on the system performance and G2Auth can work in a wide range of view angles.



(a) EER *vs.* number of waving operations.

(b) EER *vs.* number of participants.

(c) EER *vs.* resolution.

(d) EER *vs.* IMU sensor sampling rate.

(e) EER *vs.* distance.

(f) EER *vs.* illuminance level.

Figure 8: Impact of different parameters and experimental settings.

Table 3: Authentication time (and standard deviation).

Part	***without*** **pauses**	***with*** **pauses**
Authentication Operations	3250 (732.2) ms	4421 (1083.4) ms
Data Transmission	72 (8.7) ms	79 (9.8) ms
Data processing and Decision Making	34 (8.2) ms	36 (8.8) ms

7.4 Authentication Time

We measure the authentication time of G2Auth, which begins when a user waves her hand to start the authentication, and ends when a decision is made. It contains three main parts: (1) time for authentication operations; (2) time for data transmission; and (3) time for data processing and decision making (our prototype uses a cloud server to offload the computation). Time for each part is shown in Table 3. The ***total time***, without pauses and with pauses, is 3.36 ± 0.75s and 4.54 ± 1.10s on average, respectively. Thus, G2Auth can make a decision quickly.

8 USABILITY STUDY

Scanning a QR code and inputting a password are two of the most widely used authentication methods. So it helps by comparing the usability of our method against that of the two well-accepted methods, although we are aware that the two methods are insecure/inapplicable for drone delivery authentication.

8.1 Recruitment and Design

We recruit 60 subjects for this study, including 29 females and 31 males whose ages range from 15 to 68, to participate in the data collections. These subjects did not participate in our previous experiments. To avoid bias, these subjects are not informed of any method designed by us. Instead, they are told to evaluate the usability of different authentication methods.

We first ask each subject to sign a consent form and then introduce the three authentication methods. For the password-based method, we randomly generate an 8-character alphanumeric password, which is the most common length of a password [14], and show the password to the subject before authentication. For the QR code based method, a Nexus 5X smartphone, with a 5.2" screen, is used to generate and display the QR code. Next, each subject is instructed to perform five authentication attempts to get familiar with the three methods, including G2Auth. These attempts are excluded from further analysis. After that, each subject performs another three authentication attempts for each method and the order of using these methods is randomized.

After that, each subject scores the three methods by answering five questions, which are adapted from the widely-used SUS [11] to investigate the usability from the following five aspects: easy-to-use, quick, convenient, easy-to-learn, and comfortable. The five questions are listed as follows: *(1) I thought the authentication method was easy to use; (2) I am satisfied with the amount of time it took to complete the authentication; (3) I thought the authentication method was convenient; (4) I think it is easy to learn the authentication method; and (5) I felt comfortable using the authentication method.* On a scale

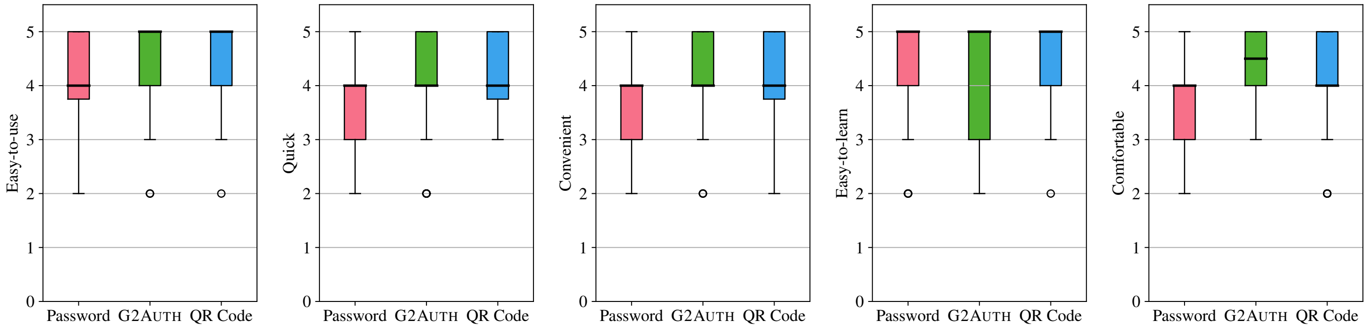


Figure 9: Usability study results.

between strongly disagree and strongly agree, the ratings for each question range from 1 to 5 (higher is better).

8.2 Usability Study Results

Figure 9 shows the results. The total scores for the password-based method, G2Auth, and the QR code-based method are 18.65 ± 2.26, 21.30 ± 2.39, 21.36 ± 2.29. The scores show that users find G2Auth and the QR code-based method have better usability than inputting an 8-character password. The difference between G2Auth and the QR code based method is small, indicating that they can achieve similar user-acceptance levels.

9 RELATED WORK

G2Auth can be categorized as *correlation-based authentication*. Many well-known systems are proposed in this direction [48, 55, 56, 60, 61]. Along the direction of correlation-based authentication, G2Auth is the first for drone delivery and has resolved many unique challenges due to the various environments and the distance between drone and user. This work is inspired by our prior work P2Auth [55]. Given a UI-constrained IoT device, which only has a button, knob or small touchscreen, to perform user authentication, a user wearing a smartwatch or carrying a smartphone does some very simple operations (e.g., clicking the button a few times). Both the IoT device and the user-side mobile device capture the timestamps about the operations and P2Auth uses the two series of timestamps to calculate a correlation score.

Unlike biometrics-based authentication [20, 34, 42, 44, 54, 73, 79, 92], G2Auth does not need to collect the user biometric information and has no concern that a user's waving habit might change over time. Given a drone-delivery task that assigns a drone D to serve a user U's order, as G2Auth compares the video recorded by D with the IMU data from U's smartphone (rather than all smartphones), its accuracy does *not* degrades as the user base grows.

Many studies are done on UAVs, such as fighting fake video timestamps [91], audio side channels [7], stolen credentials [89], and network attacks [30]. As summarized in Section 1, many patents and research works [28, 38, 67, 71, 80] have been devoted to solving the drone-delivery authentication problem. But none are resilient to relay attacks [33, 43, 68, 97]. Secure mutual authentication without special user-side infrastructure is not available prior to our work.

10 DISCUSSION

G2Auth works well under various weather conditions during our experiments (Section 7.3). We have not tested very windy or foggy weather yet. However, DJI's manual, e.g., requires "*do not use the aircraft in severe weather conditions including wind speeds exceeding 8 m/s, snow, rain, and fog*" [23]. Indeed, if the wind or fog is so heavy, the safety of drones probably becomes an issue [63]; in that case, the delivery should not be conducted in the first place.

Some users may have privacy concerns about the video recording. Such users can wear masks or cover faces using a hand. Our usability study has not received such concerns.

Compared to lockbox based authentication, G2Auth has a limitation: it requires the user to be present for package delivery. G2Auth has its advantages in other aspects: (1) It is unknown whether/when courier companies will densely install lockboxes in rural areas, while G2Auth does not depend on such infrastructure. (2) Depending on the distance of the lockbox, a user may prefer to send/receive package on her lawn than drive/walk to the lockbox in her area. (3) Some sensitive deliveries require the user's presence and signature anyway, and G2Auth can be used as a drone-based replacement for these sensitive deliveries. (4) Unless distance bounding becomes mature and widely deployed, existing lockbox solutions (such as [67]) are still vulnerable to relay attacks.

11 CONCLUSION

Authentication of drones and users for the emerging drone delivery service is an important but less-studied problem. We presented the first secure mutual authentication technique, without requiring special user-side hardware or infrastructure (i.e., only an ordinary smartphone is needed on the user side). We overcame multiple challenges, such as diverse waving styles, heterogeneous noisy data, nighttime delivery, and tracking small objects, to build an accurate and robust solution. We envision G2Auth can accelerate the deployment of drone delivery and benefit numerous users.

ACKNOWLEDGMENTS

This work was supported in part by the US National Science Foundation (NSF) under grants CNS-1856380, CNS-2016415, CNS-2107093, CNS-2144669, CNS-1953073, and CNS-1850278. The authors would like to thank the anonymous reviewers.

REFERENCES

[1] Evan Ackerman and Michael Koziol. 2019. In the Air With Zipline's Medical Delivery Drones. https://spectrum.ieee.org/in-the-air-with-ziplines-medical-delivery-drones.

[2] Aytaç Altan and Rıfat Hacıoğlu. 2020. Model predictive control of three-axis gimbal system mounted on UAV for real-time target tracking under external disturbances. *Mechanical Systems and Signal Processing* 138 (2020).

[3] Amazon. 2020. Amazon Prime Air. https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011.

[4] Apple. 2020. Getting Raw Accelerometer Events. https://developer.apple.com/documentation/coremotion/getting_raw_accelerometer_events.

[5] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. 2010. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence* 33, 5 (2010).

[6] Gildas Avoine, Muhammed Ali Bingöl, Ioana Boureanu, Srdjan Čapkun, Gerhard Hancke, Süleyman Kardaş, Chong Hee Kim, Cédric Lauradoux, Benjamin Martin, Jorge Munilla, et al. 2018. Security of distance-bounding: A survey. *ACM Computing Surveys (CSUR)* 51, 5 (2018).

[7] Adeola Bannis, Hae Young Noh, and Pei Zhang. 2020. Bleep: motor-enabled audio side-channel for constrained UAVs. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking (MobiCom)*.

[8] Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. 2009. Pearson correlation coefficient. In *Noise reduction in speech processing*. Springer.

[9] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. 2019. Learning discriminative model prediction for tracking. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[10] Stefan Brands and David Chaum. 1993. Distance-bounding protocols. In *Workshop on the Theory and Application of of Cryptographic Techniques*. Springer.

[11] John Brooke. 1996. SUS: a 'quick and dirty' usability scale. *Usability evaluation in industry* (1996).

[12] Gerard Canal, Sergio Escalera, and Cecilio Angulo. 2016. A real-time human-robot interaction system based on gestures for assistive scenarios. *Computer Vision and Image Understanding* 149 (2016).

[13] CBS News. 2020. Amazon delivery drones receive FAA approval. https://www.cbsnews.com/news/amazon-prime-air-delivery-drones-faa-approval/.

[14] Luke St Clair, Lisa Johansen, William Enck, Matthew Pirretti, Patrick Traynor, Patrick McDaniel, and Trent Jaeger. 2006. Password exhaustion: Predicting the end of password usefulness. In *International Conference on Information Systems Security*. Springer.

[15] Cas Cremers, Kasper B Rasmussen, Benedikt Schmidt, and Srdjan Capkun. 2012. Distance hijacking attacks on distance bounding protocols. In *IEEE Symposium on Security and Privacy (S&P)*. IEEE.

[16] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. 2019. Atom: Accurate tracking by overlap maximization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[17] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. 2017. Eco: Efficient convolution operators for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

[18] Martin Danelljan, Luc Van Gool, and Radu Timofte. 2020. Probabilistic regression for visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[19] Boris Danev, Heinrich Luecken, Srdjan Capkun, and Karim El Defrawy. 2010. Attacks on physical-layer identification. In *Proceedings of the third ACM conference on Wireless network security*.

[20] Alexander De Luca, Alina Hang, Frederik Brudy, Christian Lindner, and Heinrich Hussmann. 2012. Touch me once and I know it's you!: Implicit Authentication based on Touch Screen Patterns. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*.

[21] Yvo Desmedt, Claude Goutier, and Samy Bengio. 1987. Special uses and abuses of the Fiat-Shamir passport protocol. In *Conference on the Theory and Application of Cryptographic Techniques*. Springer.

[22] DHL. 2019. DHL Express Launches Its First Regular Fully-automated and intelligent Urban Drone Delivery Service. https://www.dhl.com/tw-en/home/press/press-archive/2019/dhl-express-launches-its-first-regular-fully-automated-and-intelligent-urban-drone-delivery-service.html.

[23] DJI. 2019. User Manual for Mavic Mini. https://dl.djicdn.com/downloads/Mavic_Mini/Mavic_Mini_User_Manual_v1.0_en.pdf.

[24] Yinpeng Dong, Hang Su, Baoyuan Wu, Zhifeng Li, Wei Liu, Tong Zhang, and Jun Zhu. 2019. Efficient decision-based black-box adversarial attacks on face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

[25] Saar Drimer, Steven J Murdoch, et al. 2007. Keep Your Enemies Close: Distance Bounding Against Smartcard Relay Attacks.. In *USENIX security symposium (USENIX Security)*.

[26] Dawei Du, Yuankai Qi, Hongyang Yu, Yifan Yang, Kaiwen Duan, Guorong Li, Weigang Zhang, Qingming Huang, and Qi Tian. 2018. The unmanned aerial vehicle benchmark: Object detection and tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

[27] Jeremy Elson, Lewis Girod, and Deborah Estrin. 2002. Fine-Grained Network Time Synchronization Using Reference Broadcasts. In *5th Symposium on Operating Systems Design and Implementation (OSDI)*.

[28] Thomas D Erickson, Kala K Fleming, Clifford A Pickover, and Komminist Weldemariam. 2018. Drone used for authentication and authorization for restricted access via an electronic lock. US Patent 9,875,592.

[29] Michael Esterman, Benjamin J Tamber-Rosenau, Yu-Chin Chiu, and Steven Yantis. 2010. Avoiding non-independence in fMRI data analysis: leave one subject out. *Neuroimage* 50, 2 (2010).

[30] Mohamed Amine Ferrag and Leandros Maglaras. 2019. DeliveryCoin: An IDS and blockchain-based delivery framework for drone-delivered services. *Computers* 8, 3 (2019).

[31] Sylvain Filiatrault and Ana-Maria Cretu. 2014. Human arm motion imitation by a humanoid robot. In *2014 IEEE International Symposium on Robotic and Sensors Environments (ROSE) Proceedings*. IEEE.

[32] Aurélien Francillon, Boris Danev, and Srdjan Capkun. 2011. Relay attacks on passive keyless entry and start systems in modern cars. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*.

[33] Lishoy Francis, Gerhard P Hancke, Keith Mayes, and Konstantinos Markantonakis. 2011. Practical Relay Attack on Contactless Transactions by Using NFC Mobile Phones. *IACR Cryptology ePrint Archive* 2011 (2011).

[34] Mario Frank, Ralf Biedert, Eugene Ma, Ivan Martinovic, and Dawn Song. 2012. Touchalytics: On the Applicability of Touchscreen Input as a Behavioral Biometric for Continuous Authentication. *IEEE Transactions on Information Forensics and Security* 8, 1 (2012).

[35] Davrondzhon Gafurov. 2007. A survey of biometric gait recognition: Approaches, security and challenges. In *Annual Norwegian computer science conference*. Annual Norwegian Computer Science Conference Norway.

[36] Davrondzhon Gafurov, Einar Snekkenes, and Patrick Bours. 2007. Spoof attacks on gait authentication system. *IEEE Transactions on Information Forensics and Security* 2, 3 (2007).

[37] Saurabh Ganeriwal, Ram Kumar, and Mani B Srivastava. 2003. Timing-sync protocol for sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*.

[38] Shriram Ganesh and Jose Roberto Menendez. 2016. Methods, systems and devices for delivery drone security. US Patent 9,359,074.

[39] Sinan Gezici, Zhi Tian, Georgios B Giannakis, Hisashi Kobayashi, Andreas F Molisch, H Vincent Poor, and Zafer Sahinoglu. 2005. Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks. *IEEE signal processing magazine* 22, 4 (2005).

[40] T. P. Ghuntla, H. B. Mehta, P. A. Gokhale, and C. J. Shah. 2012. A Comparative Study of Visual Reaction Time in Basketball Players and Healthy Controls. *National Journal of Integrated Research in Medicine* 3, 1 (2012).

[41] GPS.gov. 2015. GPS Accuracy. https://www.gps.gov/systems/gps/performance/accuracy/.

[42] Jun Han, Shijia Pan, Manal Kumar Sinha, Hae Young Noh, Pei Zhang, and Patrick Tague. 2017. Sensetribute: Smart Home Occupant Identification via Fusion Across On-Object Sensing Devices. In *Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys)*.

[43] Gerhard P Hancke. 2005. A practical relay attack on ISO 14443 proximity cards. *Technical report, University of Cambridge Computer Laboratory* 59 (2005).

[44] Mark R. Hodges and Martha E. Pollack. 2007. An 'Object-Use Fingerprint': The Use of Electronic Sensors for Human Identification. In *UbiComp*.

[45] Aditya Jain, Ramta Bansal, Avnish Kumar, and K. D. Singh. 2015. A comparative study of visual and auditory reaction times on the basis of gender and physical activity levels of medical first year students. *International Journal of Applied & Basic Medical Research* 5, 2 (2015).

[46] Kang Eun Jeon, James She, Perm Soonsawad, and Pai Chet Ng. 2018. Ble beacons for internet of things applications: Survey, challenges, and opportunities. *IEEE Internet of Things Journal* 5, 2 (2018).

[47] Ilchae Jung, Jeany Son, Mooyeol Baek, and Bohyung Han. 2018. Real-Time MDNet. In *European Conference on Computer Vision (ECCV)*.

[48] Nikolaos Karapanos, Claudio Marforio, Claudio Soriente, and Srdjan Capkun. 2015. Sound-Proof: Usable Two-Factor Authentication Based on Ambient Sound. In *24th USENIX Security Symposium (USENIX Security)*.

[49] Andrew J Kerns, Daniel P Shepard, Jahshan A Bhatti, and Todd E Humphreys. 2014. Unmanned aircraft capture and control via GPS spoofing. *Journal of Field Robotics* 31, 4 (2014).

[50] Felix Kreuk, Yossi Adi, Moustapha Cisse, and Joseph Keshet. 2018. Fooling end-to-end speaker verification with adversarial examples. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE.

[51] Marc Kuhn, Heinrich Luecken, and Nils Ole Tippenhauer. 2010. UWB impulse radio based distance bounding. In *2010 7th workshop on positioning, navigation and communication*. IEEE.

[52] Jennifer R Kwapisz, Gary M Weiss, and Samuel A Moore. 2011. Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter* 12, 2 (2011).

[53] Jianan Li, Xiaodan Liang, Yunchao Wei, Tingfa Xu, Jiashi Feng, and Shuicheng Yan. 2017. Perceptual generative adversarial networks for small object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

[54] Xiaopeng Li, Sharaf Malebary, Xianshan Qu, Xiaoyu Ji, Yushi Cheng, and Wenyuan Xu. 2018. iCare: Automatic and user-friendly child identification on smartphones. In *Proceedings of the 19th International Workshop on Mobile Computing Systems & Applications (HotMobile)*.

[55] Xiaopeng Li, Fengyao Yan, Fei Zuo, Qiang Zeng, and Lannan Luo. 2019. Touch Well Before Use: Intuitive and Secure Authentication for IoT Devices. In *The 25th Annual International Conference on Mobile Computing and Networking (MobiCom)*.

[56] Xiaopeng Li, Qiang Zeng, Lannan Luo, and Tongbo Luo. 2020. T2Pair: Secure and Usable Pairing for Heterogeneous IoT Devices. In *Proceedings of the ACM Conference on Computer & Communications Security (CCS)*.

[57] Lisa Johnston. 2020. What Are Webcam Frame Rates? https://www.lifewire.com/webcam-frame-rates-2640479.

[58] Yaojie Liu, Amin Jourabloo, and Xiaoming Liu. 2018. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

[59] Alan Lukezic, Tomas Vojir, Luka ˇCehovin Zajc, Jiri Matas, and Matej Kristan. 2017. Discriminative correlation filter with channel and spatial reliability. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[60] Shrirang Mare, Andrés Molina Markham, Cory Cornelius, Ronald Peterson, and David Kotz. 2014. Zebra: Zero-effort bilateral recurring authentication. In *IEEE Symposium on Security and Privacy (S&P)*.

[61] Shrirang Mare, Reza Rawassizadeh, Ronald Peterson, and David Kotz. 2018. SAW: Wristband-based Authentication for Desktop Computers. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)* 2, 3 (2018).

[62] Sjouke Mauw, Zach Smith, Jorge Toro-Pozo, and Rolando Trujillo-Rasua. 2018. Distance-bounding protocols: Verification without time and location. In *2018 IEEE Symposium on Security and Privacy (S&P)*.

[63] MavicPilot. 2018. Flying in Fog: Beware! https://mavicpilots.com/threads/flying-in-fog-beware.39412/.

[64] Daniel V. McGehee, Elizabeth N. Mazzae, and G. H. Scott Baldwin. 2000. Driver Reaction Time in Crash Avoidance Research: Validation of a Driving Simulator Study on a Test Track. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*.

[65] Yan Michalevsky, Dan Boneh, and Gabi Nakibly. 2014. Gyrophone: Recognizing speech from gyroscope signals. In *23rd USENIX Security Symposium (USENIX Security)*.

[66] Aristides Mpitziopoulos, Damianos Gavalas, Charalampos Konstantopoulos, and Grammati Pantziou. 2009. A survey on jamming attacks and countermeasures in WSNs. *IEEE Communications Surveys & Tutorials* 11, 4 (2009).

[67] Chandrashekar Natarajan, Donald R High, and V John J O'Brien. 2020. Unmanned aerial delivery to secure location. US Patent 10,592,843.

[68] Hildur Olafsdóttir, Aanjhan Ranganathan, and Srdjan Capkun. 2017. On the security of carrier phase-based ranging. In *International Conference on Cryptographic Hardware and Embedded Systems*. Springer.

[69] Payments-Next. 2019. Walmart drone delivery is up in the air. https://paymentsnext.com/walmart-drone-delivery-is-up-in-the-air/.

[70] Deepak Rajawat. 2022. 10 Best 90Hz and 120Hz Display Refresh Rate Phones To Buy In 2022. https://www.smartprix.com/bytes/best-90hz-120hz-refresh-rate-display-phones/.

[71] Soundarya Ramesh, Thomas Pathier, and Jun Han. 2019. SoundUAV: Towards Delivery Drone Authentication via Acoustic Noise Fingerprinting. In *Proceedings of the 5th Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*.

[72] Aanjhan Ranganathan and Srdjan Capkun. 2017. Are we really close? verifying proximity in wireless systems. *IEEE Security & Privacy (S&P)* (2017).

[73] Juhi Ranjan and Kamin Whitehouse. 2015. Object Hallmarks: Identifying Object Users Using Wearable Wrist Sensors. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*.

[74] Kasper Bonne Rasmussen and Srdjan Capkun. 2010. Realization of RF Distance Bounding.. In *USENIX Security Symposium (Usenix Security)*.

[75] Joseph Redmon and Ali Farhadi. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767* (2018).

[76] Wu Ren, Bo Peng, Jiefen Shen, Yang Li, and Yi Yu. 2018. Study on vibration characteristics and human riding comfort of a special equipment cab. *Journal of Sensors* 2018 (2018).

[77] Research and Markets. 2018. Drone Logistics and Transportation Market by Solution (Warehousing, Shipping, Infrastructure, Software), Sector (Commercial, Military), Drone (Freight Drones, Passenger Drones, Ambulance Drones), and Region - Global Forecast to 2027. https://www.researchandmarkets.com/research/mmcvlf/29_06_billion?w=5.

[78] RobotoLab. 2020. NAO V6 price is $9000. https://www.robotlab.com/store/nao-power-v6-educator-pack.

[79] Napa Sae-Bae, Kowsar Ahmed, Katherine Isbister, and Nasir Memon. 2012. Biometric-rich gestures: a novel approach to authentication on multi-touch devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*.

[80] Frederik Schaffalitzky. 2016. Human interaction with unmanned aerial vehicles. US Patent 9,459,620.

[81] Security Through Education. 2020. The social engineering framework: Delivery Person. https://www.social-engineer.org/framework/general-discussion/common-attacks/delivery-person/.

[82] Ivan W Selesnick and C Sidney Burrus. 1998. Generalized digital Butterworth filter design. *IEEE Transactions on signal processing* 46, 6 (1998).

[83] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K Reiter. 2016. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 acm sigsac conference on computer and communications security (CCS)*.

[84] Junjie Shen, Jun Yeon Won, Zeyuan Chen, and Qi Alfred Chen. 2020. Drift with Devil: Security of Multi-Sensor Fusion based Localization in High-Level Autonomous Driving under GPS Spoofing. In *29th USENIX Security Symposium (USENIX Security)*.

[85] Sheng Shen, Mahanth Gowda, and Romit Roy Choudhury. 2018. Closing the Gaps in Inertial Motion Tracking. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (MobiCom)*.

[86] Sheng Shen, He Wang, and Romit Roy Choudhury. 2016. I Am a Smartwatch and I Can Track My User's Arm. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*.

[87] Daniel P Shepard, Jahshan A Bhatti, Todd E Humphreys, and Aaron A Fansler. 2012. Evaluation of smart grid and civilian UAV vulnerability to GPS spoofing attacks. In *Radionavigation Laboratory Conference Proceedings*.

[88] Brian Daniel Shucker and Brandon Kyle Trew. 2016. Machine-readable delivery platform for automated package delivery. US Patent 9,336,506.

[89] Jangirala Srinivas, Ashok Kumar Das, Neeraj Kumar, and Joel JPC Rodrigues. 2019. TCALAS: Temporal credential-based anonymous lightweight authentication scheme for Internet of drones environment. *IEEE Transactions on Vehicular Technology* 68, 7 (2019).

[90] Shridatt Sugrim, Can Liu, Meghan McLean, and Janne Lindqvist. 2019. Robust Performance Metrics for Authentication Systems. In *Network and Distributed Systems Security (NDSS) Symposium*.

[91] Zhipeng Tang, Fabien Delattre, Pia Bideau, Mark D Corner, and Erik Learned-Miller. 2020. C-14: assured timestamps for drone videos. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*.

[92] Jing Tian, Chengzhang Qu, Wenyuan Xu, and Song Wang. 2013. KinWrite: Handwriting-Based Authentication Using Kinect. In *Proceedings of the 20th Network and Distributed System Security Symposium (NDSS)*.

[93] TripWire. 2017. Relay Attack against Keyless Vehicle Entry Systems Caught on Film. https://www.tripwire.com/state-of-security/security-awareness/relay-attack-keyless-vehicle-entry-systems-caught-film/.

[94] Sven Ubik and Jiří Pospíšilík. 2020. Video camera latency analysis and measurement. *IEEE Transactions on Circuits and Systems for Video Technology* 31, 1 (2020).

[95] Unmanned Systems Technology. 2020. Stabilizing Gimbals and Stabilized Camera Mounts for Drones and UAVs. https://www.unmannedsystemstechnology.com/company/gremsy/.

[96] UPS. 2020. UPS Flight Forward is changing the world of drone delivery. https://www.ups.com/us/en/services/shipping-services/flight-forward-drones.page.

[97] José Vila and Ricardo J. Rodríguez. 2015. Practical Experiences on NFC Relay Attacks with Android. In *Radio Frequency Identification*.

[98] Di Wen, Hu Han, and Anil K Jain. 2015. Face spoof detection with image distortion analysis. *IEEE Transactions on Information Forensics and Security* 10, 4 (2015).

[99] Wired. 2017. Just a Pair of These $11 Radio Gadgets Can Steal a Car. https://www.wired.com/2017/04/just-pair-11-radio-gadgets-can-steal-car/.

[100] Yi Xie, Cong Shi, Zhuohang Li, Jian Liu, Yingying Chen, and Bo Yuan. 2020. Real-time, universal, and robust adversarial attacks against speaker recognition systems. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

[101] Sangki Yun, Yi-Chao Chen, and Lili Qiu. 2015. Turning a Mobile Device into a Mouse in the Air. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*.

[102] Kexiong Curtis Zeng, Shinan Liu, Yuanchao Shu, Dong Wang, Haoyu Li, Yanzhi Dou, Gang Wang, and Yaling Yang. 2018. All your GPS are belong to us: Towards stealthy manipulation of road navigation systems. In *27th USENIX Security Symposium (USENIX Security)*.

[103] Peng Zhang, Fuhao Zou, Zhiwen Wu, Nengli Dai, Skarpness Mark, Michael Fu, Juan Zhao, and Kai Li. 2019. FeatherNets: Convolutional neural networks as light as feather for face anti-spoofing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

[104] Shifeng Zhang, Xiaobo Wang, Ajian Liu, Chenxu Zhao, Jun Wan, Sergio Escalera, Hailin Shi, Zezheng Wang, and Stan Z Li. 2019. A dataset and benchmark for

large-scale multi-modal face anti-spoofing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[105] Pengfei Zhou, Mo Li, and Guobin Shen. 2014. Use It Free: Instantly Knowing Your Phone Attitude. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking (MobiCom)*.