LightTouch: Securely Connecting Wearables to Ambient Displays with User Intent

Xiaohui Liang[1], Tianlong Yun[2], Ronald Peterson[2], and David Kotz[2]
[1]University of Massachusetts Boston and [2]Dartmouth College
xiaohui.liang@umb.edu, yuntianlong2002@gmail.com, {rapjr, kotz}@cs.dartmouth.edu

***Abstract*—Wearables are small and have limited user interfaces, so they often wirelessly interface with a personal smartphone/computer to relay information from the wearable for display or other interactions. In this paper, we envision a new method, LightTouch, by which a wearable can establish a secure connection to an ambient display, such as a television or a computer monitor, while ensuring the user's intention to connect to the display. LightTouch uses standard RF methods (like Bluetooth) for communicating the data to display, securely bootstrapped via the visible-light communication (the brightness channel) from the display to the low-cost, low-power, ambient light sensor of a wearable. A screen 'touch' gesture is adopted by users to ensure that the modulation of screen brightness can be securely captured by the ambient light sensor with minimized noise. Wireless coordination with the processor driving the display establishes a shared secret based on the brightness channel information. We further propose novel on-screen localization and correlation algorithms to improve security and reliability. Through experiments and a preliminary user study we demonstrate that LightTouch is compatible with current display and wearable designs, is easy to use (about 6 seconds to connect), is reliable (up to 98% success connection ratio), and is secure against attacks.**

I. Introduction

Wearable computers such as Google Wear smartwatches, the Nike FuelBand, and the FitBit Flex bracelet are gaining in popularity for health and fitness tracking and as companions to some smartphone applications. They are small, power-constrained, wrist or body-worn devices that integrate low-power sensors, microcontrollers and wireless interfaces to monitor physical activity and physiological body signals, notify their wearer of inbound messages, command remote computers with gestures, and more. Most such wearables are limited in their user interface. They are often paired with a personal smartphone, and used for limited functions.

We seek methods to extend wearable usage to ambient display applications. We envision a quick and effective way that captures the user's intent to initiate connection between a wearable device and an ambient display, with minimum difficulty, with minimum modifications to the display, and without assuming the wearable device has been previously introduced to the display. Our connection methods are applicable to a wide range of mobile devices, but for this paper we describe them using a bracelet as the wearable and a computer display or a smart TV as the device to connect with.

Our goal is to enable a user to use her wearable to enhance the ambient display applications. The 'ambient' display is not carried by the user, but rather is some convenient display such as a hotel television, a restaurant tablet, a treadmill monitor, or even an in-car dashboard display [1]. Using these ambient displays, the user can access and display information from her bracelet. For example, in a gym, a runner connects her mHealth bracelet to a treadmill monitor so that she is able to see her respiration rate and heart rate; in a restaurant, a customer checks her blood glucose level at a tablet before ordering food; or in a hospital, a patient connects her bracelet with a doctor's display, so they can view recent activity and health data on the display in the exam room.

Using an ambient display to access the bracelet's information extends the user's interaction with their bracelet, but it raises security concerns. Consider a scenario in which there are multiple bracelets and multiple displays present; if a user tries to wirelessly connect her bracelet to a new display, the connection may be redirected and the communication may be eavesdropped or modified by attackers. If a wrong display is connected, the private information at the bracelet may be revealed; if a wrong bracelet is connected, the user will view incorrect information at the display. Without a pre-shared secret, we require a usable method to enable the user to securely introduce her bracelet to a newly encountered ambient display. Of course, even if the devices are securely connected, the information shown on the screen could be eavesdropped by shoulder-surfing attackers. Recent studies offer methods to address such security attacks [2], [3]. In this paper, we focus on a different but more fundamental step, i.e., bootstrapping a secure RF connection.

We propose a new method to use visible light communication as a means to establish a secure RF communication channel, e.g., Bluetooth or Wi-Fi, between a bracelet and the intended display while achieving compatibility and usability. To be compatible with existing bracelets and displays, our approach uses a low-power light sensor as a light receiver at the bracelet and a desktop monitor screen as a light source at the display. A one-way brightness-modulated visible light channel (called 'brightness channel' in this paper) is then created between the screen and the light sensor. We explore a touch gesture instead of a pointing gesture, i.e., the user does not need to see the screen, but simply holds the bracelet to touch any location of the screen. (A touch-sensitive screen is not necessary.) We are particularly interested in how to achieve the secrecy and accuracy of the brightness channel and how to use these properties to improve the security of connection. Our contributions are three-fold.

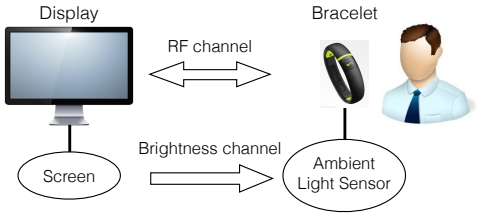


Fig. 1. System model of LightTouch

First, we introduce LightTouch, a new method of connecting bracelets to ambient displays, according to a user's intention, while achieving compatibility and usability. LightTouch does not require additional or specialized hardware in the display.

Second, our solution exploits a one-way brightness channel between the light sensor of the bracelet and the screen of the display, achieving the desired security goals. We address the secrecy problem (for which we developed a novel on-screen localization algorithm) and the error-tolerance problem (for which we developed a correlation algorithm to reduce the duration of the brightness channel communication).

Third, we implemented and evaluated LightTouch in a small, cheap, and low-power bracelet prototype and tested it using off-the-shelf displays in various ambient light conditions. The experimental results demonstrate that LightTouch achieves the desired security. With suggested parameters, LightTouch achieved 98% successful connection ratio, and the attacker could only succeed with probability 0.46% even when close to the target display. A preliminary user study showed 42 of 50 (84%) connections performed by real users were successful.

II. System model

We describe the system model and the security model.

A. System model

Figure 1 provides a high-level view of the LightTouch system. The system includes a target bracelet that belongs to a user, and an ambient display that is available for the user to use but may belong to another owner. The bracelet has a built-in ambient light sensor to detect ambient light information. The display has a screen the user can physically touch, though a touch-sensitive screen is not required. We assume the bracelet and the display have a common RF interface, such as Bluetooth, for data communication.

Consider a scenario where multiple devices are in a proximal space and they all have received a connection request from a user's bracelet. The bracelet/display needs to quickly find and create a secure communication with the display/bracelet. As indicated elsewhere, two devices cannot efficiently establish a secure communication via the RF channel alone if they have no pre-shared secret [4], [5]. LightTouch utilizes light from the display as an out-of-band channel to bootstrap the secure RF communication.

Secrecy of the brightness channel. We note that the size of the screen is often much bigger than the bracelet and the wrist: if the whole screen is used as light source, the eavesdropping attackers can access the information from the unprotected portion of the screen; however, if a fixed small portion of the screen is used as the light source, the user needs to ensure the sensor is pointed at the source, increasing the operation difficulty. LightTouch uses a novel on-screen localization algorithm to tackle this problem. The localization algorithm helps the display to derive the bracelet's location and create a location-adaptive light source. The touch gesture ensures the light source occurs underneath the bracelet and thus is only accessible to the bracelet, but not accessible to attackers. Thus, the secrecy of the brightness channel is achieved. The user-dependent source location also increases the difficulty of eavesdropping attacks because the user could use her bracelet, human wrist and human body to occlude the attacker's eavesdropping angles. As far as we know, ours is the first use of adaptive localization to enhance the secrecy of a brightness channel.

Balancing time with accuracy. The time duration of a touch gesture should be minimized as much as possible; it is impractical and error-prone if users are required to hold the bracelet still on the screen for a long time. Indeed, we found the accuracy of brightness channel is much worse than expected due to many factors, such as the screen type, the light sensor, the brightness and contrast settings of the screen, the ambient light condition, and wrist motion. Most LCD monitors support only a low refresh frequency 60-75 Hz. Some light sensors have a response delay to sudden changes in brightness. If the brightness channel lasts only for a few seconds, the sensor readings may not be accurate enough to be useful. Like other out-of-band solutions [6], efficient use of the brightness channel is a challenging goal.

B. Security model

Consider the scenario in which the user operates the connection in an insecure environment. Nearby devices manipulated by attackers may attempt to compromise the user's intention by impersonation, eavesdropping, and modification attacks. Other types of attacks, e.g., compromising the bracelet or the display, physically stealing the bracelet, and denial-of-service attacks, are beyond our scope. We assume the attackers have devices more powerful than the bracelet and the display in terms of sensing, communication, and computation capability, but may not physically touch the display.

Impersonation attacks aim to impersonate one device to establish a secure connection with the other device. In our scenario, if an attacker impersonates the display and successfully connects to the bracelet **[A1]**, the bracelet's information will be sent to the attacker; if an attacker impersonates the bracelet and connects to the display **[A2]**, the user will view incorrect information at the display. In either case, the user's intention is compromised.

LightTouch has three security goals. **[S1]** The bracelet's information is only transmitted to the display intended by the bracelet's wearer. **[S2]** The information displayed at the display is only that sent from the bracelet, and only when that bracelet's wearer has intended the information to display. **[S3]** The information from the target bracelet to the target

display cannot be eavesdropped or undetectably modified during the transmission.

III. LightTouch

We first give an overview, and then describe the algorithms.

A. Overview

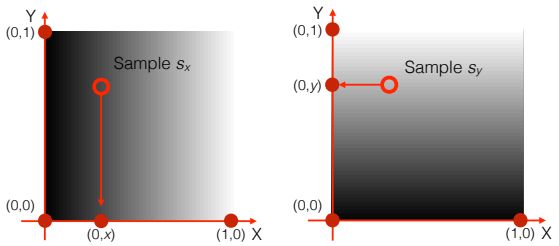
The bracelet and the display may find multiple devices in their RF communication ranges. The display's ID information can be embedded in the brightness pattern such that the wearable can identify the device with the ID information and connect to it via the RF channel. Considering an attacker can manipulate the devices with the same ID information, LightTouch is designed to help the display and the wearable verify whether the connected device is the intended device. If yes, they establish a session key according to the following four steps (shown in Table I).

Step 1. The user holds her bracelet up to any location on the screen, so that the light sensor is touching the screen. The display runs a localization algorithm to quickly derive the on-screen location L of the bracelet. Specifically, the display sends a full-screen pattern F. If the bracelet is held at any on-screen location of the display, it receives a brightness sample S. The bracelet returns S to the display via the RF channel, and the display derives the location L based on F and S.

Step 2. The display sends a hash value H of its Diffie-Hellman (DH) public parameter $g_a = g^a$ (a is the DH secret key of the display) and a freshly chosen challenge C to the bracelet. It runs an encoding algorithm to convert (H, C) into a brightness pattern. The display shows the pattern in a circular screen area centered at L with a radius r. If the bracelet is held at an actual on-screen location L' and $||L - L'|| \leq r$, it receives a sequence of brightness values corresponding to the pattern. After a decoding algorithm (separating the values and eliminating the redundant values), the bracelet obtains a sequence of encoded brightness values (H_b, C_b); these values should be correlated to (H, C) due to our encoding and decoding algorithms, but may not be identical due to environmental noise.

Step 3. The display sends its DH public parameter g_a to the bracelet via the RF channel. If the bracelet connects to the target display, it receives the correct g_a and obtains $H = h(g_a)$. The bracelet then runs a correlation algorithm to calculate the correlation score between H and H_b. Assuming the authenticity of the brightness channel, the bracelet believes H_b is from the target device and only accepts H if the correlation algorithm outputs 1. The intuition of using correlation is to tolerate some noise. If the bracelet accepts H, it proceeds to the next step; otherwise, it stops the protocol.

Step 4. The bracelet encrypts C_b with session key $sk = g^{ab}$ and sends the ciphertext $\mathcal{E}$ and its DH public parameter $g_b = g^b$ to the display (b is the DH secret key of the bracelet). Only the display knowing secret a can calculate the session key $sk = h(g_b^a)$ and obtain $C_b = Dec(sk, \mathcal{E})$. If the display finds that C_b is correlated to C, it confirms the connection; otherwise, it stops the protocol. The intuition is that C can only be received by the target bracelet (assuming the secrecy of the brightness channel), and therefore, receiving a correlated C_b means the display is connected to the target bracelet.



(a) F_x (b) F_y

Fig. 2. Full-screen patterns

After finishing the four steps in the bootstrapping protocol, the display confirms the connection. It then encrypts a message 'success' with session key sk, and sends the ciphertext $\mathcal{D}$ to the bracelet. The bracelet uses the session key sk to decrypt $\mathcal{D}$. If it is 'success', the bracelet notifies the wearer of the successful connection.

B. Localization algorithm

As mentioned above, we allow the user to place her bracelet at any location on the screen, for convenience. Ours is the first work that uses such a localization algorithm to enhance the secrecy of a brightness channel.

We design a pattern F shown on the full screen. The bracelet, placed at any on-screen location, is able to capture a sample of F. F is a sequence of four distinctive full-screen images $\{F_{min}, F_{max}, F_x, F_y\}$: F_{min} is filled with a dark gray scale $(\eta_{min}, \eta_{min}, \eta_{min})$ (RGB value), F_{max} is filled with a lighter gray scale $(\eta_{max}, \eta_{max}, \eta_{max})$, F_x is filled with fine-grained gray scales (η, η, η) where pixel brightness η increases monotonically from η_{min} to η_{max} along the X-axis as shown in Figure 2(a); and F_y is filled with the same gray scales, which increase monotonically along the Y-axis as shown in Figure 2(b). The bracelet is supposed to receive four readings $S = (s_{min}, s_{max}, s_x, s_y)$ corresponding to four images respectively, and send S back to the display via the RF channel. The display receives S, calculates the derived location $L = (x, y)$ based on F and S, and then shows the secret pattern in a circular screen area centered at L with a radius r. The display sets r to an appropriate value, not too small (so the bracelet can receive the secret pattern) and not too large (to ensure the bracelet and the wrist can cover and protect the light source). We discuss the grayscale ranges and the calculation of L and r in Section IV.

C. Encoding algorithm

The display runs an encoding algorithm to convert a secret number into a pattern. A secret pattern is a sequence of grayscale values. The number of used grayscale values is a constant k, which is known to both the display and the bracelet. The grayscale values (from the most dark to the most bright) are denoted by $b_1, \cdots, b_k$ where $b_1 = (\eta_{min}, \eta_{min}, \eta_{min})$ and $b_k = (\eta_{max}, \eta_{max}, \eta_{max})$. The difference between any two adjacent grayscale values is $\frac{\eta_{max} - \eta_{min}}{k-1}$.

TABLE I
NOTATION AND LIGHTTOUCH BOOTSTRAPPING PROTOCOL

F	full-screen pattern	$\dashrightarrow_F$	full-screen light source	Enc, Dec	symmetric key
L	bracelet's location	$\dashrightarrow_L$	light source at L		encryption and decryption
S	brightness samples	$\rightarrow, \leftarrow$	RF channel	sk	session key
Cr	correlation algorithm	L_b, t_b	bracelet's correlation parameters	h	secure hash function
C	secret challenge	L_d, t_d	display's correlation parameters	g, a, b	Diffie-Hellman parameters

	Display ($g_a = g^a$)		**Bracelet** ($g_b = g^b$)
① Localization	Send a full-screen pattern F	$\dashrightarrow_F$	Receive S at one location L
	Receive S, and derive location L	$\leftarrow$	Send S
② En- & De-coding	Send $H = h(g_a)$ and C	$\dashrightarrow_L$	Receive H_b, C_b
③ Correlation	Send g_a	$\rightarrow$	Receive g_a If $Cr(h(g_a), H_b, L_b, t_b) = 1$, g_a is verified. Calculate $sk = h(g_a^b)$
④ Correlation	Receive $\mathcal{E}$ and g_b, and calculate $sk = h(g_b^a)$ If $Cr(C, Dec(sk, \mathcal{E}), L_d, t_d) = 1$, g_b and connection are verified.	$\leftarrow$	Send $\mathcal{E} = Enc(sk, C_b)$ and g_b
	Send $\mathcal{D} = Enc(sk,$ 'success')	$\rightarrow$	If 'success' = $Dec(sk, \mathcal{D})$, connection is verified.

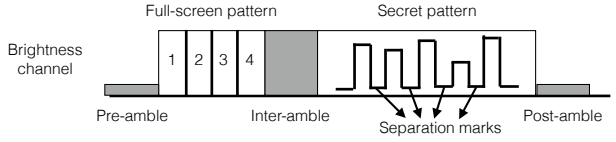


Fig. 3. Encoding and decoding for the Brightness Channel

To encode H, the display converts H into a β-length k-ary number $h_1 \cdots h_\beta$ where $h_j \in [0, k-1]$ for $1 \leq j \leq \beta$. For each h_j, the display then finds the grayscale value b_{h_j} and maps H to a brightness pattern $(b_{h_1+1}, \cdots, b_{h_\beta+1})$. To encode C, the display uses the same method.

D. Decoding algorithm

The display and the bracelet adopt a special decoding algorithm. The display adds recognizable signals to the pattern such that the bracelet can decode the pattern from the sensor readings by using these signals. As shown in Figure 3, the pre-amble, inter-amble, and post-amble are signals added to help the bracelet to detect when and what patterns are transmitted. The pre-amble is used to indicate that the full-screen pattern is about to be sent, and the post-amble is used to indicate that the whole transmission is ended. The inter-amble is used to indicate that localization is finished and the secret pattern is about to send. The pre/post-amble are set as black, while the inter-amble is set as white. All three are displayed for a relatively long time T_a to be recognizable.

The display shows the secret pattern at frequency $1/T_b$, i.e., each grayscale value is shown for duration T_b. LightTouch requires the receiving frequency at the light sensor to be higher than $1/T_b$ such that each grayscale value has at least one sensor reading. However, a higher receiving frequency may cause repeated sensor readings for one grayscale value. LightTouch uses separation marks to solve the problem. Each separation mark is added between any two grayscale values in the full-screen pattern and the secret pattern as shown in Figure 2. The separation mark is set as black with time period T_f shorter than T_a so it is distinguishable from the minimum grayscale value η_{min} and the pre/inter/post-amble.

Adding separation marks enables the bracelet to separate the sensor readings for adjacent grayscale values. Since most light sensors are basically resistors, they do not respond instantly to the changes in screen brightness. The display may also have a performance delay of showing the patterns. Finally, the display and the bracelet must synchronize time using the separation marks. The bracelet considers the series of sensor readings, one window $\mathcal{W}$ at a time. We set the window width to match the signal period, T_b+T_f. If the number of readings during T_f is N_f, and the number of readings during T_b is N_b, the window has $N = N_f + N_b$ readings. The bracelet finds the maximum reading within the N-reading window; if the middle value of the peak-detection window is the peak value, $\mathcal{W}$ is shifted forward by $\lfloor N/2 \rfloor$, i.e., the $\lfloor N/2 \rfloor$ oldest values in the window are replaced with $\lfloor N/2 \rfloor$ newest ones; if not, $\mathcal{W}$ is shifted by 1, i.e., the oldest value in the window is replaced with the new one. In this way, the grayscale value is extracted (the peak value in the window) and the window quickly becomes aligned with the signal. The length N is set so that $\lfloor N/2 \rfloor \leq N_f$ to capture the peak value if it appears at the end of T_b, and $\lfloor N/2 \rfloor \geq N_b$ to avoid capture of two repeated peak values for one grayscale value. As such, we have $N_b \leq \lfloor N/2 \rfloor \leq N_f$. If the sensor uses a constant frequency, we have $T_f \geq T_b$. To improve efficiency, we choose the minimum $T_f = T_b$, i.e., $N_b = N/2 = N_f$.

We used a similar approach (separation marks and peak detection) for transmitting and receiving the full-screen pattern. Since the localization accuracy determines the effectiveness and the security of LightTouch, we prefer to choose T_b' and T_f' for showing the full-screen pattern larger than T_b and T_f. We choose $T_b' = T_f'$ and $T_b = T_f$. The time duration we use the brightness channel in LightTouch is thus $T = 3T_a + 8T_b' + 2(2\beta - 1)T_b$.

E. Correlation algorithm

We prefer to use an error-tolerant correlation algorithm rather than a bit-extraction algorithm because: i) the sensor readings are not accurate enough for bit-extraction; and ii) the bracelet and the display can leverage their RF channel.

The bracelet receives a sequence of brightness values from the display, and obtains H_b and C_b after the decoding process above. It also obtains g_a from the display via the RF channel

TABLE II
EXPERIMENT PARAMETERS.

Bracelet: Arduino UNO board	Display: 24" HP monitor
Light sensor: 3 Photocells	h: SHA-256 hash function
BLE: nRF8001	H, C, sk: 256-bit number

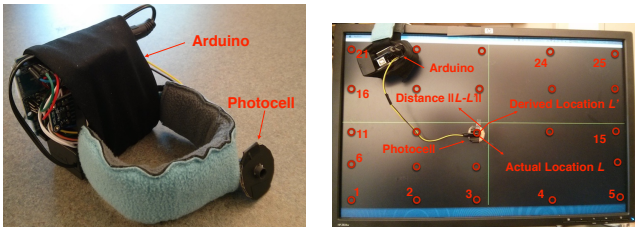
and calculates $H = h(g_a)$. The correlation algorithm with inputs (H, H_b, L_b, t_b) outputs 1 if i) the correlation score is $< L_b$ and ii) the difference of their lengths is $< t_b$, and outputs 0 otherwise. (L_b, t_b) are adjustable parameters. If we choose a larger L_b or t_b, the bracelet tolerates more differences between H_b and H (high usability), and the chance for the impersonation attacks to succeed is greater (low security). The correlation function begins by normalizing H_b and H, resulting in $\bar{H}_b$ and $\bar{H}$. Because H and H_b might not be of equal length, we use a dynamic time warping algorithm [7] to calculate the correlation score of $\bar{H}_b$ and $\bar{H}$. In this algorithm, the two sequences are "warped" non-linearly in the time dimension to determine a measure of their similarity independent of certain non-linear variations in time.

After the bracelet sends C_b to the display, the display uses a similar correlation algorithm with parameter (L_d, t_d) to verify that C and C_b are correlated. The balance between usability and security is the focus of the correlation algorithm, which we analyze in Section IV.

IV. PERFORMANCE EVALUATION

Security, compatibility, and usability are our three goals for LightTouch. Before evaluating the algorithms, we first check the compatibility and usability of LightTouch from a high-level view. For compatibility, we implemented the LightTouch bracelet using a popular Arduino board equipped with an ambient light sensor (photocell) and a Bluetooth Low Energy (BLE) module; we implemented the LightTouch display as a Python program compatible with MacOS, Windows, and Linux operating systems. The RF channel between the bracelet and the display is implemented using BLE, which is widely supported by existing bracelets, smartphones, tablets, laptops, and computers due to its low power consumption. (We expect it will soon be implemented in many smart TV and similar ambient display devices.) We attempted to evaluate our solutions by using commercially-available wrist devices but we found that programmable wrist devices restrict the highest sensing frequency of the ambient light sensor to 1Hz; it appears this limit is imposed by the wearable operating system rather than the hardware itself. Instead, we built a bracelet prototype with cheap photocells where the sensing frequency is more easily adjusted. To evaluate usability, we tested LightTouch with three photocells, one at a time. LightTouch achieved an average 98% successful connection ratio while resisting impersonation attacks against the display and the bracelet, an encouraging result. The parameters are shown in Table II.

We first evaluated the accuracy of the localization algorithm and the impersonation attacks that use the leaked patterns. We then studied the efficiency of the encoding and decoding algorithms. Last, we evaluated the effectiveness of the correlation algorithm and the impersonation attacks that guess the DH public parameters.



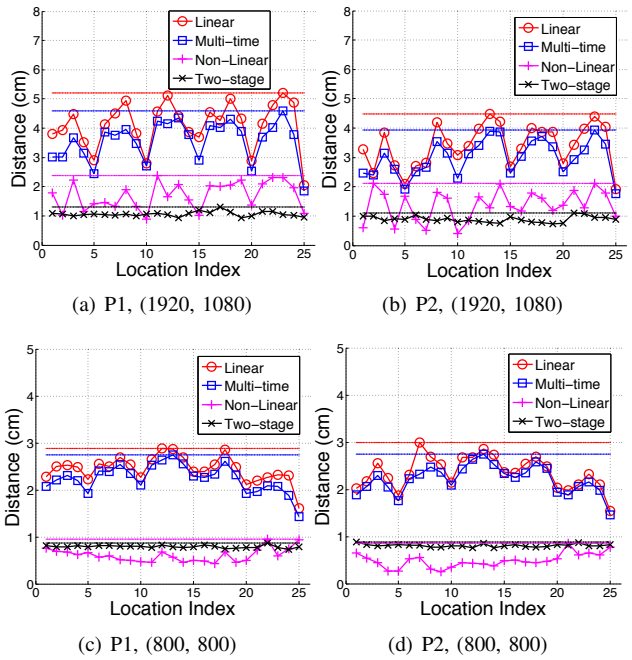
(a) Bracelet prototype (b) Localization setup
Fig. 4. Bracelet and Display Setup

Accuracy of localization: The localization algorithm helps the display to calculate a derived bracelet's location L', which could be different from the bracelet's actual location L. In practice, the display is not able to obtain L or $\delta = ||L - L'||$. As such, the display has to choose a radius r large enough to ensure the photocell can access the light source regardless of δ. We use the following settings for evaluating localization: we evaluate the full-screen pattern by using two different sized windows (1920,1080) and (800, 800) on an HP monitor. The actual size of (1920,1080) is 51.8 cm$\times$ 29.1 cm and that of (800,800) is 21.5 cm $\times$ 21.5 cm. Note that the "full-screen" pattern does not necessarily take up the whole screen. We set the display to max brightness and max contrast to maximize the difference between any two grayscale values. We set $T'_b = 100$ ms and $T_a = 150$ ms. We chose 25 locations uniformly distributed over the defined screen for placing the photocells as shown in Figure 4(b). For each location, the display uses a linear interpolation of $(s_{min}, s_{max}, s_x, s_y)$ to derive the location:

$$L = (x, y) = (x_{max} \cdot \frac{s_x - s_{min}}{s_{max} - s_{min}}, y_{max} \cdot \frac{s_y - s_{min}}{s_{max} - s_{min}})$$

We recorded the 25 measured locations $L' = (x', y')$, and calculated each distance $\delta = ||L - L'|| = ((x-x')^2 + (y-y')^2)^{1/2}$. The maximum distances δ corresponding to the 25 locations are shown by the flat horizontal lines in Figure 5 where each value is the maximum of 100 localization runs. In the linear interpolation case we found that the distance δ was the largest when the photocell was near to the central point of the full-screen pattern. In addition, though the photocells have a small sensing surface, they may receive multiple grayscale values. This could be another cause of localization errors. From our experimental results as shown in Figure 5, if the display relies on the one-time linear interpolation to derive L', r needs to be about 4-5 cm for $(1920, 1080)$, and 2-3 cm for $(800, 800)$. If we put the photocell at the center of a circular paper board, to cover the light source, the paper board needs to have a radius $2r$, i.e., 8-10 cm for $(1920, 1080)$ and 4-6 cm for $(800, 800)$ as shown in Figure 6. Accordingly, the required width, i.e., the diameter of the paper board, is 16-20 cm or 8-12 cm.

From the National Library of Medicine [8], the average human wrist circumference of women at height (5'2"-5'5") and men at height ($>$ 5'5") are 15.24 cm and 16.51 cm, respectively, and the corresponding wrist widths are approximately



(a) P1, (1920, 1080) (b) P2, (1920, 1080)

(c) P1, (800, 800) (d) P2, (800, 800)

Fig. 5. Evaluation of four localization methods

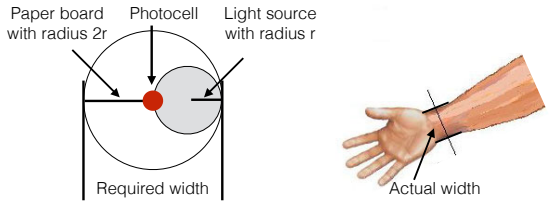


Fig. 6. Required width and actual width

4.8 cm and 5.2 cm. We also examined existing smart watches, most of which have a 4 cm $\times$ 4 cm screen. The required widths 16-20 cm or 8-12 cm are much larger than the actual width of a bracelet and human wrist. As such, this one-time linear localization was not good enough. We further improved the localization accuracy by using the following methods.

1) Multi-time method: If the display shows the full-screen pattern multiple times, the photocell will capture multiple samples. The display could obtain the average values of the samples and derive a more precise location. As shown in Figure 5, if other parameters are the same, the multiple-time method generally outperformed the one-time localization algorithm and reduces r by 0.23-0.61 cm. A disadvantage of this method is that it increases the duration of the brightness channel communication, i.e., the time required for users to hold their bracelets up to the screen.

2) Two-stage method: The idea for this method is to adaptively change the size of the screen pattern in two stages. In the first stage, the display shows the full-screen pattern at full size, e.g., (1920, 1080). Suppose the display obtains a derived location L' and based on the linear method it knows the maximum localization error is r_1. In the second stage, the display shows a smaller full-screen pattern centered at L' with a size $(2r_1, 2r_1)$. This guarantees the smaller pattern can be seen by the photocell. After receiving the second sample corresponding to the smaller pattern, the display derives a more precise location of the bracelet. From Figure 5, the two-stage method significantly improves the localization accuracy compared to the one-time and the multi-time linear methods. With the two-stage method, r can be chosen around 1 cm or even smaller than 1 cm in some cases. The two-stage method doubles the time needed for localization and keeps the features of the linear interpolation which can be easily implemented. One disadvantage of this method is it requires learning the statistics of r_1 and chooses r_1 conservatively to ensure successful readings of the photocell in the second stage.

3) Non-linear method: Readings from photocells do not have a strictly-linear relation with grayscale value η for many reasons. Displays commonly apply a Gamma correction to colors to compensate for properties of human vision because human eyes have greater sensitivity to relative differences between darker tones than between lighter tones [9]. The goal of Gamma correction is to maximize the use of the bandwidth relative to how humans perceive light and color. In our experiments the light captured by the photocell from the screen has been altered by the Gamma correction Γ. So we tried learning such changes for different grayscale values and using that to reverse the effects of Gamma correction. We ran tests of localization at 25 locations, 100 runs at each, and based on the measured distance we learned a reverse Gamma correction function $\bar{\Gamma}$. We added $\bar{\Gamma}$ as the last step of calculating the locations of the bracelet. As shown in Figure 5, the performance of the non-linear method was comparable with the two-stage method for $(800, 800)$, while it was faster, requiring only transmission of one full-screen pattern. For pattern size $(1920, 1080)$ the non-linear method outperformed the one-time and multi-time methods, but it was not as good as the two-stage method.

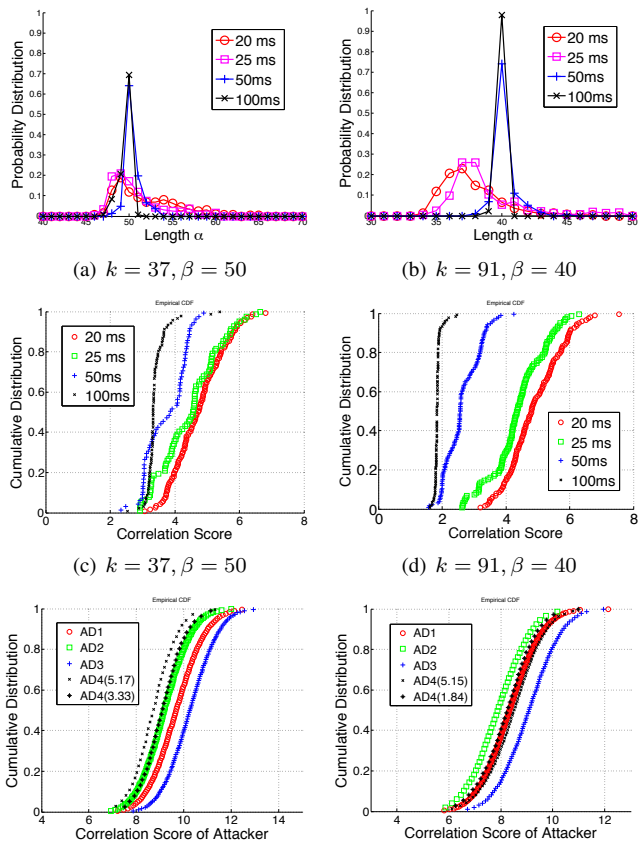
Impersonation attacks using the leaked patterns. Attacks A1 and A2 may benefit from information leaked when the display transmits a secret pattern. We experimented with the two-stage and non-linear localization methods. We chose photocell P2, and screen size (800,800). We made three round-shaped and non-transparent paper boards, with radius r_b = 1 cm and 1.25 cm (shown in Figure 4). We fixed photocell P2 at the center of the boards, and used transparent sticky tape to fix the boards on the screen. To simulate the different locations of the photocell, we generated the full-screen pattern at random locations on the screen, while ensuring the photocell was located inside the full-screen pattern. After the localization algorithm completes, it showed the secret pattern centered at the derived location L' with a radius r for 3 seconds, and then launched the next round of localization. For the experimental results shown in Figure 5(f), we chose $r = 0.69$ cm if the display used the non-linear method and $r = 0.45$ cm if the display used the two-stage method. We repeated the localization test for 500 rounds for each setting, and used a camera to record the whole screen of the monitor. Here, one camera was enough; as shown in Figure 4(b), the camera can

cover the whole screen. In addition, the ambient light sensor with a board tightly touches the screen so that no side angles can be used to access the brightness pattern underneath the board. Thus, the camera could identify localization failures if the pattern was shown at other on-screen locations. The recording time for a test of each setting was about 6-8 minutes. We later scanned the video: if we visibly saw a secret pattern extending beyond a board, we added 1 to a localization failure count. For the two-stage method, we found the board with $r_b = 1$ cm was big enough to cover the light source with $r = 0.45$ cm. We did not see the secret patterns in any case. On the other hand, for the non-linear method, we found 230 localization failures out of 500 runs when $r_b = 1$ cm and 41 failures out of 500 runs when $r_b = 1.25$ cm. The two-stage method provides much better accuracy on localization, and thus a better secrecy of the brightness channel than the non-linear method. As such, LightTouch prefers to use the two-stage method for localization.

Efficiency of encoding and decoding: The efficiency and accuracy of the pattern transmission over the brightness channel depends on two parameters: i) the time duration T_b for showing a grayscale value in the brightness pattern and ii) the length β of the pattern (as a k-ary number). We chose $T_b = 20$ or 25 ms for a test with high efficiency (readings are not stable) and chose $T_b = 50$ or 100 ms for a test with high accuracy. The grayscale value range was set as $[60, 240]$. By setting the increment of η as 5 or 2, we obtained $k = 37$ or $k = 91$ grayscale values, respectively. We used a SHA-256 hash function h to generate $H = h(g_a)$. The 256 bits of H are converted to a k-ary number to be sent over the brightness channel. Based on the above parameter settings, if $T_b = 20$ ms twenty-five 37-ary numbers can be transmitted via the brightness channel in one second. The size of the number space in one second is about $37^{25} \approx 2^{130}$. In other words, 130 bits can be transmitted per second, and the time needed to transmit a 256-bit number is 1.96 s. We realize that existing approaches could achieve much higher data-transmission rates, e.g., a barcode approach can achieve throughput of 91-172 kbps [10], and the LED approach uses a high frequency (1000 Hz) signal to transmit megabits per second [11]. However, they are not compatible with our scenario where the display screen only supports 60-75 Hz refresh rates, and a camera-based solution was not suitable for a bracelet due to its size and power limitations.

Effectiveness of correlation: In LightTouch, the correlation algorithm has two steps: comparing the length α of H_b with the length β of H (as a k-ary number), and correlating H_b with H. Thus, our evaluation has two steps as well:

1) In Figures 7(a) and 7(b), we plot the probability distribution of the length α of H_b. All of the probability distributions in this experiment are obtained from at least 250 samples. We can see that the relatively slow change of the screen brightness enabled the photocells to capture H_b of a similar length to H. When $T_b = 50$ or 100 ms, the range of the number of peaks was $[47, 53]$ for $\beta = 50$ and $[38, 45]$ for $\beta = 40$. In comparison, when $T_b = 20$ or 25 ms, the ranges extended



(a) $k = 37, \beta = 50$ (b) $k = 91, \beta = 40$

(c) $k = 37, \beta = 50$ (d) $k = 91, \beta = 40$

(e) $k = 37, \beta = 50$ (f) $k = 91, \beta = 40$

Fig. 7. Evaluation of correlation

to $[46, 65]$ for $\beta = 50$ and $[34, 49]$ for $\beta = 40$. From the distribution, we see that the photocell obtained H_b with a length similar to H. The bracelet can set $t_b = 8$ for $\beta = 40$ and $t_b = 10$ for $\beta = 50$ to accept more than 99% of samples.

2) In Figures 7(c) and 7(d), we plot the cumulative distribution of the correlation score between H_b and H. We used 1-norm distance in the dynamic time-warping algorithm. Generally, if T_b decreased, the probability of having a large correlation score increased: if the transmission of one grayscale value was short, such as 20 ms or 25 ms, the screen and photocells both generated more errors. In Figures 7(c) and 7(d), when $\beta = 50$, we chose L_b to be $(3.9, 4.6, 6.0, 6.1)$ and when $\beta = 40$, we chose L_b to be $(2.1, 3.5, 5.7, 6.3)$, respectively for $T_b \in \{100, 50, 25, 20\}$ ms. With these L_b and t_b values, LightTouch achieved 98% successful connection ratio.

Impersonation attacks guessing eligible DH public parameters: Consider an attacker A1 that uses a device to impersonate the display's RF communications and connect to the target bracelet. The attacker also has a device connecting via RF to the display but not touching the display, and the attacker uses this device to generate fake samples to ensure that the bracelet is able to receive H_b and C_b from the display via the brightness channel (otherwise, the attack can be detected by the bracelet). The attacker is able to know g_a (sent by the display over the RF channel), $H = h(g_a)$, the length

β of H (as a k-ary number), t_b and L_b, but it does not know H_b because the secrecy of the brightness channel is achieved. Then, the attacker forges a DH public parameter g'_a and sends g'_a to the bracelet via the RF channel in Step 3. Denote the length β' of $H' = h(g'_a)$ (as a k-ary number). To succeed, the attacker needs to ensure $\beta' \in [\beta - t_b, \beta + t_b]$ and the correlation score of H' and H_b is smaller than L_b. The attacker can generate as many DH public parameters as it wants. However, the attacker cannot choose a specific hash value H' and reversely obtain the parameter g'_a. Thus, H' is a random number from the attacker's perspective. A1 can test g'_a in many ways and selectively send g'_a to the bracelet. To get g'_a pass the verification, A1 can adopt four good strategies (other strategies have less success probability):

- AD1 sends g'_a to the bracelet if $\beta' \in [\beta - t_b, \beta + t_b]$;
- AD2 sends g'_a to the bracelet if $\beta' = \beta - t_b$;
- AD3 sends g'_a to the bracelet if $\beta' = \beta + t_b$;
- AD4 sends g'_a to the bracelet if $\beta' \in [\beta - t_b, \beta + t_b]$ and the correlation score of (H', H) is $<$ a threshold (set as 6).

In our experiment, the attacker first randomly chooses 10^6 DH public parameters and obtains 10^6 corresponding hash values H'. The attacker then converts the hash values into k-ary numbers, and only chooses the numbers with lengths in $[\beta - t_b, \beta + t_b]$ because the attacker knows other hash values would not be accepted by the bracelet. We chose $T_b = 20$ ms, calculated the correlation scores of these hash values with H_b, and plot the cumulative distribution in Figure 7(e) and 7(f). For $k = 37$ (Figure 7(e)), we see that the most effective strategy was AD4 (who carefully used the reference H and its length β). Especially when the correlation score of H and H_b is small, e.g., 3.33 and 5.17, AD4's hash value H' is much more likely to be correlated with H_b because H' is correlated with H and H is correlated with H_b. On the other hand, for $k = 91$ (Figure 7(f)), we found the most effective strategy was AD2, i.e., choosing length-32 hash values. When $k = 91$, β' was the major factor in the correlation score while using the value of H is not helpful.

Now, let us look closer at the successful connection ratio and successful attack probability together. When $T_b = 20$ ms, the bracelet has to set $L_b = 6.3$ to achieve 98% successful connection ratio for $k = 37$ and $k = 91$, while AD4 (5.17) succeeds with a probability 0.46% for $k = 37$ and AD2 succeeds with a probability 8.7% for $k = 91$. Note that the attacker cannot increase its success probability by simply repeating the protocols because the devices reset the parameters every time after the correlation fails. The hash value H_b used for correlation is different for each run. One reason that the correlation when $T_b = 20$ ms and $k = 91$ could not distinguish a forged H' from the authentic H effectively is we chose too many grayscale values, i.e., $k = 91$. In fact, the grayscale values ranged from 60 to 240, which means the grayscale value increased by 2 per level. Therefore, the noise from the screen and the photocells are overwhelming factors to the readings. We conclude that, to achieve both usability and security, LightTouch should use $k = 37$ to define the grayscale values. In this case, LightTouch achieved 98% successful connection ratio and the attacker can only succeed with probability 0.46% (using strategy AD4).

Time efficiency: The time needed to send a full-screen pattern is 800 ms (100 ms $\times$ 4 $\times$ 2) if the non-linear method is used, and 1600 ms if the two-stage method is used. The shortest time for the brightness channel to send a secret pattern is 1980×2 ms (one for H and one for C) when $k = 37$, $\beta = 50$ and $T_b = 20$ ms. We set pre/inter/post-ambles to 150 ms. In summary, the time for the brightness channel is $150 \times 3 + 800 + 1980 \times 2 = 5210$ ms (with non-linear localization) or $150 \times 3 + 1600 + 1980 \times 2 = 6010$ ms (with two-stage localization). The process takes about six seconds, which we believe is quite acceptable as the first prototype.

Preliminary user study: We conducted a preliminary user study of LightTouch. Five users each repeated the LightTouch connection process ten times. We used the parameters suggested in the previous section; the connection succeeded 42 times out of 50 attempts. In contrast to our benchtop successful connection ratio 98%, the users succeeded only 84% of the time. The motion of the user's wrist could cause the light sensor to move to a different location and lose parts of transmitted patterns. We added an accelerometer to the bracelet prototype. We found that the acceleration data collected during 7 of the failed connections indicate an observable movement of the user's wrist. We envision multiple ways to increase the successful connection ratio and thereby improve the usability of LightTouch: i) increase the size of the light source to allow for slight wrist movement, ii) change the correlation parameters, and iii) reduce the time needed for the protocol. These options all trade off security and usability, and require a more extensive user study to fully evaluate human factors and LightTouch usability in real-world settings (varying displays, devices, sensors, environment, users, and attacks).

V. Related work

Proximity is often considered as a means for expressing intent of connecting two devices at a short distance. In proximity, the audio signals [12], [13], [14] and radio signals [15], [16] received at two devices exhibit a similar pattern, which can be used as a secret. NFC and RFID are short-range radio communication technologies, and they can also be used to send a secret from one device to another. In proximity-based approaches, attackers are assumed to be physically away from the target devices. However, this assumption might not be true in a mobile environment. Some security-enhanced techniques have been proposed: some researchers developed an attenuation technique enabling users to adaptively change the NFC communication range [17], and others explored a specific user gesture as an additional user intent [18]. As proximity alone does not represent user intent, these solutions require user effort to compensate.

Passkey-based approaches require the user's visual, gesture, or memory effort to input the same key in both devices or to pass the key from one device to the other [19], [20], [21]. The user's intent is confirmed if the same passkey is input to two devices by users. If both devices have keypads or one has a

keypad and the other has a screen, this approach is easy to implement; Apple's AirPlay is one example of this approach, which can connect a MacOS computer with an Apple TV. However, for interface-constrained devices, such as bracelets, a screen and keypad may not be available. GesturePIN proposed ten gestures including 3D stroke-based directional movements as the passwords [20]. Users could wear the bracelet and perform the gestures according to the instruction shown on the screen, and the acceleration data will be decoded as the passkey. However, such approach requires user effort, is error-prone, and is subject to visual eavesdropping attacks.

Correlating sensor data is another approach for expressing user intent. ShakeWellBeforeUse requires a user to shake two devices at the same time such that the acceleration data sensed separately at the devices can be correlated and the acceleration data can be used as secret or used for mutual authentication [22]. ZEBRA continuously monitors the behavior of user's computer interaction, and correlates the user's wrist movement captured by the bracelet with the inputs from keyboard and mouse of the used computer [23]. In general this method helps two devices with sensing capability obtain user intent from the same user's behavior. In our scenario, we seek solutions for standard displays with no specialized hardware for sensing.

VI. Conclusions

In this paper, we introduce LightTouch, a novel distributed communication approach that enables users to securely connect their bracelets with ambient displays. Our approach respects a user's intentionality in a compatible, usable, and secure solution, in which a screen and ambient light sensor are used to create an out-of-band channel for bootstrapping secure communication. We developed a novel on-screen localization method with a touch gesture and a location-adaptive light source, so our method successfully resists impersonation attacks on the establishment of a secure connection. We used a pattern correlation approach (with a tunable balance between accuracy and efficiency) to resist display impersonation attacks. LightTouch is compatible with existing wearables due to its use of low-cost, low-power sensors, and is usable with minimum user effort, i.e., a touch gesture; LightTouch is secure because it ensures the communication link corresponds to user intent. We implement LightTouch in a small, cheap, and low-power bracelet prototype and test it using off-the-shelf displays. Through our experiments conducted in a lab setting, we show that LightTouch interaction can complete around 6 seconds, while connecting successfully in 98% of test cases and achieving a low attack probability of 0.46%.

VII. Acknowledgements

This research program is supported by National Science Foundation award number CNS-1329686. The views and conclusions in this document are those of the authors and may not necessarily represent the official policies of NSF.

References

[1] *The International Symposium on Pervasive Displays (PerDis) Copenhagen, Denmark*. ACM, June 2014.

[2] S. Iizuka, W. Naito, and K. Go, "A study for personal use of the interactive large public display," in *International Conference on Human Interface and the Management of Information*, 2013, pp. 55–61.

[3] F. Brudy, D. Ledo, S. Greenberg, and A. Butz, "Is anyone looking? mitigating shoulder surfing on public displays through awareness and protection," in *Proceedings of the International Symposium on Pervasive Displays (PerDis)*, 2014, pp. 1–6.

[4] J. M. McCune, A. Perrig, and M. K. Reiter, "Seeing-is-believing: Using camera phones for human-verifiable authentication," in *IEEE Symposium on Security and Privacy*, 2005, pp. 110–124.

[5] B. Zhang, K. Ren, G. Xing, X. Fu, and C. Wang, "SBVLC: Secure barcode-based visible light communication for smartphones," in *Proceedings of the Annual IEEE International Conference on Computer Communications (INFOCOM)*, 2014, pp. 2661–2669.

[6] R. Kainda, I. Flechais, and A. W. Roscoe, "Usability and security of out-of-band channels in secure device pairing protocols," in *Proceedings of the Symposium on Usable Privacy and Security (SOUPS)*, 2009.

[7] F. Petitjean and P. Gançarski, "Summarizing a set of time series by averaging: From Steiner sequence to compact multiple alignment," *Theoretical Computer Science*, vol. 414, no. 1, pp. 76–91, Jan. 2012.

[8] National Library of Medicine, "Calculating body frame size," http://www.nlm.nih.gov/medlineplus/ency/imagepages/17182.htm, 2012.

[9] C. Poynton, *Digital Video and HDTV Algorithms and Interfaces*, 1st ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2003.

[10] T. Hao, R. Zhou, and G. Xing, "COBRA: color barcode streaming for smartphone systems," in *International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2012, pp. 85–98.

[11] S. Schmid, G. Corbellini, S. Mangold, and T. R. Gross, "LED-to-LED visible light communication networks," in *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2013, pp. 1–10.

[12] D. Schürmann and S. Sigg, "Secure communication based on ambient audio," *IEEE Transactions on Mobile Computing*, vol. 12, no. 2, pp. 358–370, 2013.

[13] C. Soriente, G. Tsudik, and E. Uzun, "HAPADEP: Human-assisted pure audio device pairing," in *Proceedings of the International Conference on Information Security (ISC)*, 2008, pp. 385–400.

[14] M. T. Goodrich, M. Sirivianos, J. Solis, G. Tsudik, and E. Uzun, "Loud and clear: Human-verifiable authentication based on audio," in *IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2006, pp. 1–10.

[15] L. Cai, K. Zeng, H. Chen, and P. Mohapatra, "Good neighbor: Ad hoc pairing of nearby wireless devices by multiple antennas," in *Network and Distributed System Security Symposium (NDSS)*, 2011.

[16] S. Mathur, R. D. Miller, A. Varshavsky, W. Trappe, and N. B. Mandayam, "ProxiMate: proximity-based secure pairing using ambient wireless signals," in *International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2011, pp. 211–224.

[17] R. Zhou and G. Xing, "nShield: A noninvasive NFC security system for mobile devices," in *International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2014, pp. 95–108.

[18] A. Czeskis, K. Koscher, J. R. Smith, and T. Kohno, "RFIDs and secret handshakes: defending against ghost-and-leech attacks and unauthorized reads with context-aware communications," in *ACM Conference on Computer and Communications Security (CCS)*, 2008, pp. 479–490.

[19] S. N. Patel, J. S. Pierce, and G. D. Abowd, "A gesture-based authentication scheme for untrusted public terminals," in *Proceedings of the Annual ACM Symposium on User Interface Software and Technology (UIST)*, 2004, pp. 157–160.

[20] M. K. Chong, G. Marsden, and H. Gellersen, "GesturePIN: using discrete gestures for associating mobile devices," in *Proceedings of the International Conference on Human Computer Interaction with Mobile Devices and Services (Mobile HCI)*, 2010, pp. 261–264.

[21] J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan, "uWave: Accelerometer-based personalized gesture recognition and its applications," *Pervasive and Mobile Computing*, vol. 5, no. 6, pp. 657–675, 2009.

[22] R. Mayrhofer and H. Gellersen, "Shake well before use: Intuitive and secure pairing of mobile devices," *IEEE Transactions on Mobile Computing*, vol. 8, no. 6, pp. 792–806, 2009.

[23] S. Mare, A. Molina-Markham, C. Cornelius, R. Peterson, and D. Kotz, "ZEBRA: Zero-effort bilateral recurring authentication," in *Proceedings of the IEEE Symposium on Security and Privacy*, 2014, pp. 705–720.