LightDefender: Protecting PIN Input using Ambient Light Sensor

Jiacheng Shang and Jie Wu
Center for Networked Computing, Temple University, Philadelphia, PA 19121

***Abstract*—Nowadays, personal identification number (PIN) is one of the most popular methods for identity verification. However, recent researches show that attackers can easily recover victims' PINs in spite of the large number of combinations PIN provides. Existing protection approaches require alteration of the original interaction between the user and PIN-based authentication systems, or still fail if the attacker can observe and mimic the victim's input behavior. Considering these limitations, we propose a defense system called LightDefender to protect current PIN-based systems from PIN replay attacks using a single ambient light sensor. Specifically, we protect the PIN input by leveraging the biometrics in the received light intensity that is influenced by input behaviors and biological features. To our best knowledge, our work is the first one to protect PIN input using the light intensity. Different from existing approaches, LightDefender does not change the original interaction methods between the user and PIN-based authentication systems, and the extra hardware cost is low. In addition, by leveraging biological differences (e.g. finger length) among different users, LightDefender still claims high-security protection against strong attackers who can mimic the victim's input behaviors. Experiments with 10 volunteers show that LightDefender can achieve an average true acceptance rate of 95% for normal users. More importantly, LightDefender can correctly reject two types attackers with an average true rejection rate of at least 93.6% without data of new attackers.**

***Index Terms*—Personal identification number, ambient light, PIN input protection.**

I. Introduction

User authentication is an important procedure for a system to verify the identity of the user. Among all authentication approaches, personal identification number (PIN) is one of the most popular ones because of its combination of both usability and security. A PIN is a numeric or alpha-numeric password used in the process of authenticating a user accessing a system. A 4-digit PIN can result in 10,000 possible combinations. Therefore, PIN is widely used to withdraw cash from an ATM, unlock a mobile device, open a door, and so on. However, recent researches show that attackers can easily recover the victims' PINs in spite of a large number of combinations PIN provides. The attacks can be grouped into three categories. First, to achieve good usability, users tend to pick context-related PINs (e.g. birthday), which largely decreases the randomness of PINs and makes it easier for the attacker to hit the PIN [1]. Second, the attacker can perform shoulder-surfing attacks or leverage a camera to record the victim's input procedure [2]–[6]. Third, recent researches show that the attacker can use various sensor information and side channel (e.g. acceleration and acoustic signals) to help in recovering victims' PINs [7]–[10]. Therefore, it is essential to have a defense system that adds protection to PIN-based authentication systems, especially if the PIN has been leaked.

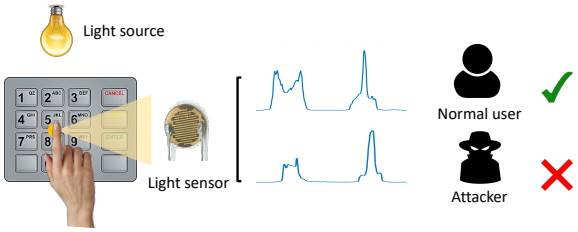


Fig. 1. LightDefender system scenario.

To defend PIN users against potential threats of PIN leakage, most existing approaches focus on preventing the attacker from acquiring the victim's PIN, and they can be classified into two categories: challenge-response-based approaches [11]–[13] and indirect input-based approaches [14]–[16]. In challenge-response-based approaches, the user is asked to input the correct response that is calculated using the PIN based on a random challenge. However, by repeating the challenge procedure, the attacker can still gather useful information about the original PIN based on multiple challenge-response pairs [17], [18]. To address this issue, various solutions are proposed to prevent the attacker from observing the challenge-response pairs by using secure secondary channels [19]–[28]. However, these approaches suffer from low usability and high learning cost. Similar to the challenge-response-based approaches, indirect input systems ask users to input on a secondary interface . However, indirect input-based approaches still alter the original interaction methods. To ensure the good usability of the defense system, other researchers propose to defend against PIN leakage by leveraging the input behavior (e.g. velocity and direction) during PIN input [29]. However, since they only consider simple time-domain features (e.g. velocity magnitude and directions), they fail to defend against the attackers who can mimic the victim's input behavior through shoulder-surfing and recording attacks [30].

Considering the limitations of existing approaches, we propose a defense system that aims to protect the current PIN-based authentication systems from PIN replay attacks. A defense system should meet three key requirements. First, the defense system should have high usability, which means that it cannot significantly change the original interaction methods between the user and the PIN-based authentication systems. Second, the defense system should provide high-security protection to PIN input against strong attackers who can mimic the victim's input behavior. Third, the extra hardware cost should be as low as possible. In this paper, to meet the above requirements, we propose a new system called LightDefender to defend against PIN replay attacks using a

single ambient light sensor. In PIN replay attacks, an attacker already has the victim's PIN via some way and aims to break PIN-based authentication systems by inputting the victim's PIN. Specifically, we protect the PIN input by leveraging the biometrics in the received light intensity that is influenced by input behaviors and biological features (e.g. finger length). Different from existing approaches, LightDefender does not change the original interaction method between the user and PIN-based authentication systems, and the extra hardware cost is low. In addition, compared with input behavior-based approaches, LightDefender provides protection against strong attackers who can mimic the victim's input behaviors.

As shown in Fig. 1, LightDefender consists of two major components: a low-cost light sensor and a light source. The ambient light sensor lies in the center of the PIN pad or keyboard and converts the received light intensity to the output voltage. The light source is over the light sensor and continuously emits visible light. When a user inputs a PIN, the palm and fingers block partial incident light, which generates different light intensities received at the light sensor. Therefore, even if an attacker can "replay" the victim's PIN to authentication systems, the light intensity signal is different from that of the victim as long as the attacker does not follow the victim's input behavior. Moreover, even if an attacker can record and mimic the victim's input behavior by performing shoulder-surfing and recording attacks, the received light intensity of the attacker is still distinct from that of the victim because of biological differences (e.g. finger length and width) between the attacker and the victim. These biological differences also introduce variances to the received light intensity. In this paper, we investigate the possibility of protecting PIN input using the biometrics in the raw output voltage of a single ambient light sensor, while eliminating the influences of noise (e.g. movement of nearby people). In particular, we develop a mechanism to detect the fine-grained starting and ending points of the PIN input only based on the raw output voltage signals. Then, we extract 34 features from the raw output voltage signals in the time domain, the frequency domain, and the time-frequency domain. These features are used to build a classification model that is used to determine whether the input is from the normal user. Our contributions are as follows:

- Our work serves as a feasibility assessment to show that the light intensity influenced by the PIN input contains rich biometric information and can be used to verify the identity of the user. To our best knowledge, our system is the first to use ambient light to protect the PIN input.
- We propose a mechanism to accurately detect the starting and ending point the PIN input by analyzing the raw output voltage signals. In total, 34 features are extracted and used to build a multiple additive regression tree-based classification model for the final decision.
- We develop a prototype and conduct comprehensive evaluations. Experiments with 10 volunteers show that LightDefender can achieve an average true acceptance rate of 95% for normal users. Moreover, LightDefender can correctly reject two types of PIN replay attackers with an average true rejection rate of at least 93.6% even if no data of new attackers is available .

II. Related work

PIN leakage. Although PIN is proposed as an authentication method with high security, recent researches show that it can be reconstructed using various techniques. In general, these attacks can be grouped into three categories: statistics-based approaches [1], camera-based approaches [2]–[6], and side-channel information-based approaches [7]–[10]. For statistics-based approaches, a recent study shows that the most common numbers follow some patterns and tend to be based on some context (e.g. birth date) [1]. In camera-based approaches, the attacker can reconstruct the PIN with high accuracy based on the video-based side-channel information. Moreover, recent researches show that sensors in the victim's mobile and wearable devices can reveal its sensitive PIN [7]–[9]. Other works are also proposed to infer the PIN by using various side-channel information (e.g. acoustic signals) [6], [8], [10].

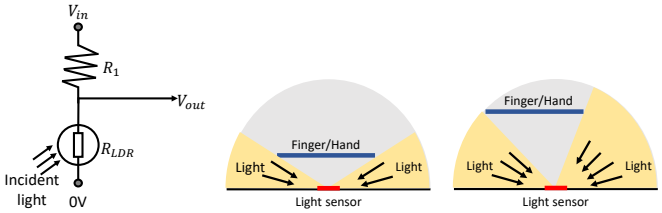
Defence against PIN leakage. Considering the threats of PIN leakage, various systems are designed to either prevent the attacker from acquiring the victim's PIN. These systems can be further classified into two categories: Challenge-response-based approaches [11]–[13], [19]–[28] and Indirect input-based approaches [14]–[16]. Challenge-response-based approaches [11]–[13] are all based on the insight that the attacker who does not know the mapping function cannot recover the victim's PIN based on the newly constructed password. However, by repeating the challenge procedure, the attacker can still gather useful information of the original PIN based on multiple challenge-response pairs [17], [18]. To solve this problem, various solutions are proposed by delivering the random challenge through various secure secondary channels that are invisible to the attacker [19]–[28]. Although their approaches achieve high accuracy on defending against shoulder-surfing attackers, they usually come with low usability and introduce extra cost to users for learning the new system.

Similar to the challenge-response-based approaches, indirect input systems prevent the attacker from observing the PIN input procedure by leveraging a secondary input interface. However, indirect input-based approaches still alter the original interaction method of the PIN input, and the secondary interfaces usually introduce high hardware cost (e.g. Google glass). There are also many systems that try to authenticate the user by leveraging the biometrics in input behaviors or keystrokes [29], [31]–[33]. However, they only consider simple features mainly in the time domain such as velocity magnitude and device acceleration. It is still possible for an attacker to perfectly mimic the victim via shoulder-surfing attacks [30].

III. Preliminary

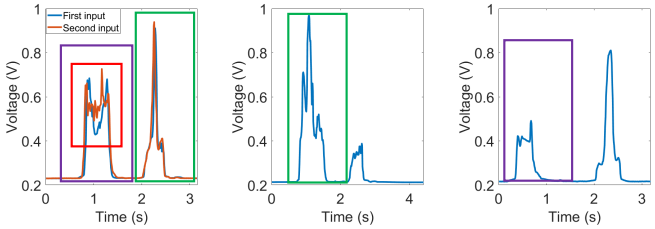
A. Ambient light sensor

A light sensor generates an output signal indicating the intensity of light by measuring the radiant energy that exists in a very narrow range of frequencies basically called "light",



(a) A circuit diagram of LDR-based light sensor. (b) 2-D illustration of incident light when the finger moves vertically and horizontally.

Fig. 2. Human vocal system and two propagation paths of the voice.



(a) Normal user. (b) Simple replay attack (c) Strong replay attack

Fig. 3. The output voltage of light sensor when the normal user and two types of attackers input the same PIN.

and which ranges in frequency from "Infra-red" to "Visible" up to "Ultraviolet" light spectrum. Among all types of light sensors, the photoconductive cell using Light Dependent Resistor (LDR) is the most common. The LDR is made from a piece of exposed semiconductor material that changes its electrical resistance based on the received light intensity. Fig. 2(a) shows a circuit diagram of an LDR-based light sensor. We can acquire the light intensity level by measuring the voltage V_{out} at their junction. The output voltage V_{out} is determined based on:

$$V_{out} = V_{in} \times \frac{R_{LDR}}{R_{LDR} + R_1}, \quad (1)$$

where V_{in} and V_{out} are input and output voltage, respectively. R_1 is a series resistor, and R_{LDR} is the light dependent resistor. When the light intensity is low, the resistance of the light dependent resistor reaches a high value, which produces a high output voltage. In contrast, the output voltage is low when the received light intensity is high. Since the value of R_{LDR} will never be zero or infinity, the LDR sensor is expected to measure the light intensity from 0 to infinity.

In our system, we embedded an LDR-based light sensor in the middle of a keyboard. The underlying principle of light-based gesture recognition systems [34]–[36] and our light-based defense system are fundamentally similar: the hand can reflect or block the light, which further influences the received light intensity at the sensor. The light is from a fixed light source that is over the PIN pad (e.g. attached on the shield) and emits lights with consistent intensity. Fig. 2(b) shows how the vertical and horizontal movements of the finger influence the incident light on a 2-D plane. The yellow region illustrates the space in which the light can reach the ambient light sensor. The finger and hand are modeled as a line. We can observe that, if the finger moves away from the light sensor vertically, more light reaches the light sensor, which produces higher light intensity. Similarly, if the finger moves away from the light sensor horizontally, the received light intensity rises since more direct light reaches the light sensor. Moreover, due to the biological differences (e.g. finger length, finger width, and palm size) among different users, the received light intensities are also different even if the fingers and hands of two users are at exactly the same location.

B. Attack model

In our attack models, the attacker aims to break PIN-based authentication systems (e.g. ATM machine) by "replaying" the victim's PIN to authentication systems. The capability of the attacker is limited in the sense of:

A simple PIN replay attack. In this attack model, the attacker can acquire the victim's PIN by using non-vision techniques (e.g. motion sensors). Therefore, the attacker only has the victim's PIN without knowing how the victim inputs the PIN. To break the PIN-based authentication system, the attacker inputs the PIN with a random input behavior (e.g. different fingers).

A strong PIN replay attack. In this type of attack model, we assume that the attacker can use vision-based techniques (e.g. a hidden camera) to infer the victim's input behavior, which means that the attacker knows not only PIN but also the victim's input behavior. To break the PIN-based authentication system, the attacker inputs the PIN while imitating the victim's input behavior.

C. Feasibility study

To validate our observations, we build a sensing platform by embedding an LDR-based ambient light sensor in the middle of a keyboard. A light-emitting diode (LED) bar is installed over the keyboard to act as the major light source so the user's hand and fingers can block the light as long as they are over the keyboard. The feasibility experiments are done in an office room where multiple light sources exist. The details of the platform setup are shown in Section VII. We first asked a volunteer to input a six-digit PIN ("146928") twice, and the measured output voltage signals are shown in Fig. 3(a). We can observe that the user's input behavior introduces much greater influences to the raw output voltage measurements than other factors (e.g. the activities of other people in the same room). Moreover, the output voltage patterns of the same user are consistent overall. Although the same user cannot perfectly reproduce the same pattern (small variations in the red box) still exist, we can still extract useful knowledge (e.g. overall shape and frequency) from the raw output voltage to match the patterns from the same user.

Moreover, we first asked the simple replay attacker to input the victim's PIN in its own way on the same testbed, and the raw output voltage is shown in Fig. 3(b). We can see that the output voltage pattern is distinctive from that of the victim because their input behaviors (e.g. finger used and habitual hand) are different. Especially in the second phase of the input behavior (green box), the peak-to-peak distance of the victim's data is much higher than that of the attacker's data. We also

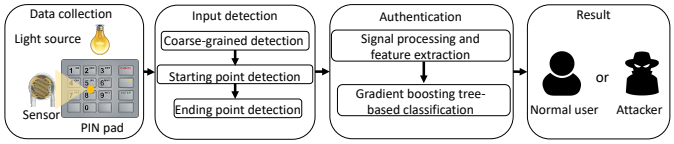


Fig. 4. System pipeline.

collected data from the strong replay attacker, as shown in Fig. 3(c). We used a camera to record the whole process of the victim's PIN input. The strong replay attacker is required to watch the video until it is confident to imitate the victim's input behavior. We can see that the strong replay attacker is able to produce a similar voltage pattern to the victim in terms of the overall shape, but the amplitude is still significantly distinct from that of the victim. In the first stage of the input behavior (purple box), the average amplitude of the victim's data is about 0.5 V, while that of the strong replay attacker is about 0.4 V. The reason behind this is the biological differences between the victim and the attacker. For example, even if two different fingers are at the same location, different finger length and width determine the amount of light that is blocked, which produces different light intensity received at the light sensor. Therefore, even if the strong replay attacker can perfectly imitate the victim's input behavior, it cannot produce the same voltage pattern as long as its hand and fingers are biologically different from those of the victim.

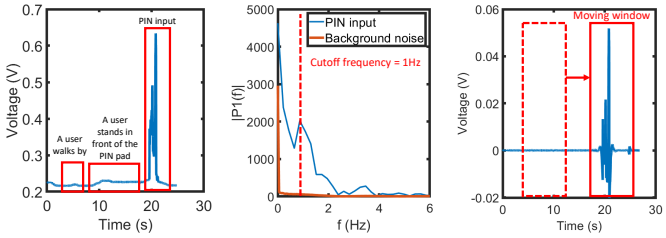
D. Challenges

Fine-grained input detection. In order to defend against the PIN replay attack using an ambient light sensor, we first need to extract the sensor signal that is influenced by the PIN input. In general, the PIN input procedure consists of three phases: moving hands over the keyboard, inputting the PIN, and moving hands back. A naive solution is to acquire the key pressing time from the PIN-based authentication system, but this approach only reserves the sensor signal in the second phase while losing all information in the other two phases. To extract the sensor signals that contain the information in all three phases, we proposed an energy-based input detection approach based on the insight that PIN input has greater influences on the sensor values than other factors.

Accurate classification model using proper features. After getting the raw output voltage data of the whole PIN input procedure, we need to extract features that are consistent for the same user and distinctive between the user and the attacker. Moreover, the classification model should be robust to the collinearity of extracted features because features are heterogeneous across different domains. To address this problem, we extract features from the time domain, frequency domain, and time-frequency domain of the raw output voltage signal. To leverage the collinearity of features from three domains, we use a multiple additive regression tree for classification.

IV. System Overview

We build a system that mainly contains two major phases: the enrollment phase and the authentication phase. The processes of both phases follow the pipeline shown in Fig. 4.



(a) The raw output voltage signal. (b) Fast Fourier transform of the raw signal. (c) The output signal of high-pass filter.

Fig. 5. Analysis of the output voltage signal.

Enrollment phase. In the enrollment phase, the user is asked to repeat inputting its PIN several times. Since the user is not able to give the accurate starting and ending time of the PIN input, LightDefender processes the raw signal to extract the output voltage signal that is influenced by the PIN input. Considering the frequency of PIN input is at least 1 Hz, LightDefender first removes the influence of background noise by filtering the output voltage signals through a high-pass filter and detects the coarse-grained location of the PIN input by studying the short-time energy of filtered signals. Then, LightDefender detects the accurate starting and ending time of the PIN input by analyzing the short-time energy around the coarse-grained location based on a threshold. The extracted output voltage signals are used to extract features that can represent the identity of the user. These features are trained together with attackers' data (collected in advance) in the database to build a strong classifier.

Authentication phase. After collecting enough training data from the user, the system is ready to be used for authentication. The system can be used by the normal user or an attacker. For each authentication attempt, we first detect and extract the PIN input in the same way as in the enrollment phase. After that, we extract the same 34 features of the new input and send it to the multiple additive regression tree-based model. An attacker is detected and rejected if the classification model recognizes the new input as from an attack.

V. PIN Input Detection

A. Coarse-grained PIN input detection

To defend against PIN replay attackers using the ambient light sensor, we first need to accurately detect the starting and ending time of the PIN input behavior. In general, the PIN input behavior can be segmented into three stages: 1) Moving the hand over the PIN pad; 2) Inputting PIN; 3) Moving the hand back. A simple solution is to acquire the pressing time of the first and last keys from the authentication systems. However, this solution will lose the sensor information in the first and the third stages. Moreover, the output voltage of the ambient light sensor is also constantly influenced by other factors (e.g. human activities nearby), which makes it hard to detect the starting and ending points using a threshold.

To address this issue, instead of directly detecting the fine-grained starting and ending points, we first find the coarse-

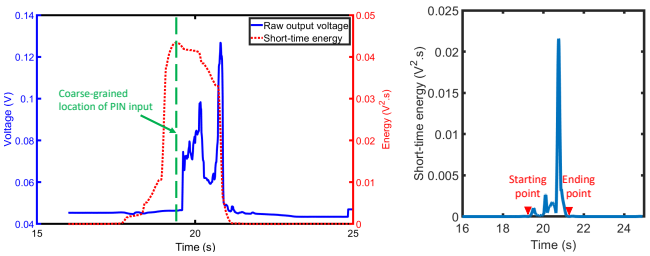
grained location of PIN input gesture in the noisy output voltage signals using the finding algorithm in [37]. The accurate starting and ending points are then detected around the coarse-grained locations. Fig. 5(a) shows the raw output voltage when a user walks to a PIN authentication system and inputs a PIN. It is clear that the PIN input behavior will introduce much greater influence than other human activities. Moreover, as shown in Fig. 5(b), most background noise has a frequency of less than 1Hz, while the PIN input still has information with a frequency of larger than 1 Hz. Based on these two observations, we find the coarse-grained location of PIN input by analyzing the short-time energy of noisy output voltage signals. In order to remove the pulses caused by background noise, we apply a 3-order high-pass butter filter on the raw signals with a cut-off frequency of 1 Hz, and the filtered signal is shown in Fig. 5(c). We can see that pulses introduced by the PIN input are much more significant in the filtered output voltage signal. To further remove the pulses caused by background noise, we apply a threshold filter on the output of the high-pass filter. The threshold is set as the mean of the high-pass filter's output, excluding the highest 40% and the lowest 40% of the measurements. All measurements whose values are lower than the threshold will be 0 after passing through the threshold filter. To find the coarse-grained location of the PIN input, we apply a moving window to the filtered output voltage signals and compute the short-time energy within each window. The window size is set to 1.8 seconds in our system for two reasons. First, it is the minimal time to input a 6-digit PIN. Second, by using the minimal time as the window size, we can ensure that the signal within the window is only influenced by PIN input. Since the pulses introduced by the PIN input are much more significant in the filtered output voltage signal, the starting point of the moving window must be within the PIN input procedure when the short-time energy within the window reaches the highest value. Therefore, the coarse-grained location of the PIN input can be detected by solving:

$$\arg\max_{s} ([g_s, g_{s+1}, \ldots, g_{s+w}])([g_s, g_{s+1}, \ldots, g_{s+w}])^T, \quad (2)$$

where s is the coarse-grained location of the PIN input, $G = [g_1, g_2, \ldots, g_n]$ is the filtered output voltage signal, n is the length of the filtered signal G, w is the size of the moving window, and $([g_s, g_{s+1}, \ldots, g_{s+w}])([g_s, g_{s+1}, \ldots, g_{s+w}])^T$ computes the short-time energy of the window starting from the s^{th} sample to the $(s+w)^{th}$ sample. Fig. 6(a) shows the short-time energy of windows starting from different samples. We can see that the short-time energy reaches its highest value at 19.4 seconds, which is exactly during the PIN input.

B. Fine-grained starting and ending points detection

Since the detected coarse-grained location lies within the procedure of PIN input, the accurate starting and ending points must show near the coarse-grained location. Moreover, we find that the output voltage values are pretty stable before and after the PIN input because the user will not move before and right after the PIN input. Therefore, the values of these two stable stages should be close to zero after



(a) Coarse-grained detection. (b) Fine-grained detection.

Fig. 6. PIN input detection.

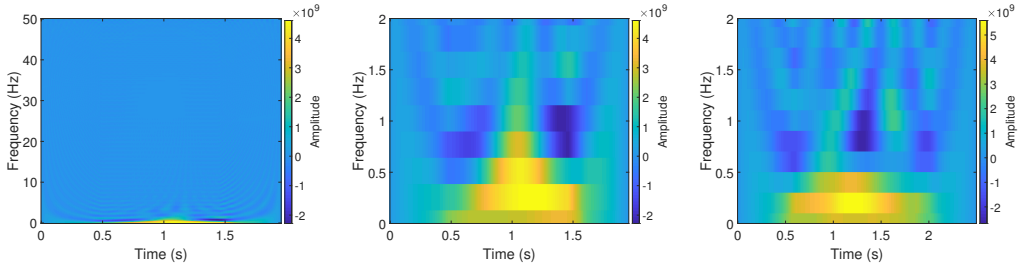
filtering the raw output voltage signal with high-pass and threshold filters. This observation enables us to detect the accurate starting and ending points by checking short-time energy changes before and after the coarse-grained location. To detect accurate starting and ending points of PIN input, we first apply a moving window on the filtered signal in coarse-grained location detection. To achieve better granularity, we set the window size to 0.3 seconds, and the result is shown in Fig. 6(b). We can see that the short-time energy is very low (close to zero) until the PIN input starts. Since the most noise is removed in the filtered signal, we can accurately detect the starting and ending time by finding the first and the last points whose energy exceeds a threshold around the coarse-grained location. In our testbed, the threshold is set to 0.00001 $V^2.s$, and both starting and ending points should be within 5 seconds from the detected coarse-grained location.

VI. User Authentication

A. Feature extraction

To train a strong classifier that can detect the replay attacker, we need to extract useful features from output voltage signals that are influenced by the PIN input. Here, useful features are those that are consistent for the same user but distinctive between the normal user and the attacker. In our system, we select 34 different features from the time domain, the frequency domain, and the time-frequency domain.

Features in the time domain. We extract six features from the time domain, including the maximum, the average amplitude, peak-to-peak distance, variance of the signal, root-mean-square (RMS) level, and the average dynamic time wrapping (DTW) distances between the new data and the templates that are selected from the normal user's pre-collected data. Specifically, the maximum, the average amplitude, and peak to peak distance describe the overall amplitude of the raw output voltage, which is mainly influenced by biological features such as finger length and width. The RMS level and variance are used to describe the trend of the signal. The DTW distance is used to measure whether the new data has a similar shape as the user's template that is collected during the enrollment phase. Since we only consider the overall shape of the detected output voltage signal, we normalize each output voltage signal individually over the range of the ADC to eliminate the influence of voltage value. Moreover, we smooth each raw output voltage signal using a moving average filter with a window length of 20 samples.



(a) The Wigner-Ville distribution of the signal influenced by the PIN input. (b) The low-frequency Wigner-Ville distribution of the victim. (c) The low-frequency Wigner-Ville distribution of the strong attacker.

Fig. 7. The Wigner-Ville distribution of the victim and the strong attacker.

Feature in the frequency domain. To capture the features in the frequency domain, we perform a fast Fourier transform (FFT) on the extracted output voltage signals. Six features are extracted from the FFT result, including skewness, kurtosis, mean value, median value, variance, and peak-to-peak distance. These features describe the rhythm of how the user presses the key and blocks the incident light.

Feature in the time-frequency domain. Besides extracting features from the time and frequency domains individually, we also study how the PIN inputs influence the output voltage signal in each time and frequency frame. We first apply maximal overlap discrete wavelet transform (MODWT) using the Haar wavelet down to the fourth level on the raw signal and perform multiresolution analysis on the MODWT matrix. The reason we choose MODWT rather than classic discrete wavelet transform is that MODWT can achieve translation-invariance by removing the downsamplers. We extract the mean value, peak-to-peak distances, RMS, and variance from the results of the multiresolution analysis as features. Also, we calculate the Wigner-Ville distribution of the raw signal. Compared to a short-time Fourier transform, the Wigner-Ville distribution function can furnish higher clarity. For a discrete signal $G = [g_1, g_2, \ldots, g_n]$ with n samples, the Wigner-Ville distribution is defined as:

$$WVD_G(t, f) = \sum_{k=-n}^{n} G(t + \frac{k}{2}) G^*(t - \frac{k}{2}) e^{\frac{-j2\pi fk}{n}}, \quad (3)$$

where t is the time vector, f is the frequency vector, and $G^*(t - k/2)$ is the complex conjugate of $G(t - k/2)$. Fig. 7(a) shows the Wigner-Ville distribution of the output voltage signal influenced by the PIN input. We can observe that the PIN input influences the output voltage signal mainly in the low-frequency bands. Therefore, we further check the Wigner-Ville distribution of the voltage signals influenced by PIN inputs of the normal user and the strong replay attacker respectively, and the results are shown in Figs. 7(b) and 7(c). It is clear that the energy distribution is distinctive in the low-frequency bands in two aspects. First, the entries with the lowest amplitude appear at different locations in two Wigner-Ville distributions. As shown in Fig. 7(b), in the user's distribution, the entry with the lowest amplitude is at the later stage of the PIN input. While in the strong attacker's distribution, the entry with the lowest amplitude appears at the middle stage of the PIN input. Second, the energy distribution in each frequency band is very distinctive between the user and strong replay attacker, which means that we can detect the attacker by checking the standard deviation of the energy distribution in each frequency frame. Therefore, we extract the location of the minimal amplitude and its amplitude value as three features. Moreover, we calculate the standard deviation of the energy distribution for each frequency frame under 2 Hz and include them into the feature vector. To deal with different frequency resolutions caused by different lengths of signals, we resize each Wigner-Ville distribution to the same size so that the first 15 frequency frames exactly cover the frequency range from 0 Hz to 2 Hz.

B. Classification based on multiple additive regression tree

To determine whether the extracted features are from the normal user or any type of PIN replay attackers, we train a binary classifier based on Multiple Additive Regression Tree (MART). Compared with other machine learning models, the gradient boosting-based approach has three major advantages. First, MART is robust to various types of features with different scales and units, which exactly exists in our feature vectors. For example, the value of the maximal amplitude is in the range from 0 to 1, while the values of DWT features can be less than 0.001. Second, features extracted from different domains may not be totally independent of each other. By using MART, the classifier can effectively deal with the colinearity of features across various domains.

The basic idea of MART is to build a strong classifier using a set of weak classifiers. Different from other gradient boosting approaches, MART specializes the gradient boosting approach to the case where each weak classifier is a regression tree. Here we use the formulation of MART in [38]. After M rounds, the estimation $F(x)$ of the strong classifier is an additive expansion of the form

$$F(\mathbf{x}) = \sum_{m=0}^{M} b_m h(x; \mathbf{a}), \quad (4)$$

where $h(\mathbf{x}; \mathbf{a})$ is a weak classifier with parameters $\mathbf{a} = \{a_1, a_2, \ldots, a_K\}$ and feature vector $\mathbf{x} = \{x_1, x_2, \ldots, x_N\}$. In each iteration, the coefficients b_m and the parameters $\mathbf{a}_m$ are jointly fit to the training data in a forward "stage-wise" manner. Starting with an initial guess $F_0(x)$, the coefficients b_m and the parameters $\mathbf{a}_m$ in the m^{th} iteration can be found by solving the following problem:

$$(b_m, \mathbf{a}_m) = \arg\min_{b, \mathbf{a}} \sum_{i=1}^{N} L(y_i, F_{m-1}(\mathbf{x}_i) + bh(\mathbf{x}_i; \mathbf{a})), \quad (5)$$

where y_i is the diagnosis variable, and $L(y, F)$ is the loss function that is used to define lack-of-fit. Therefore, the

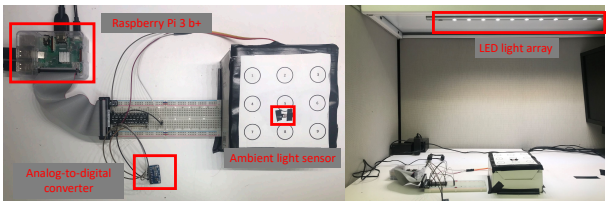


Fig. 8. Testbed.

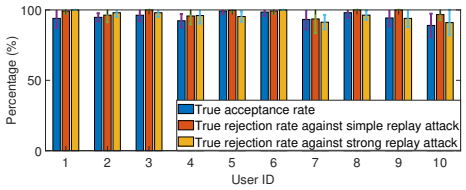


Fig. 9. The overall performance.

estimation of the strong classifier after the m^{th} iteration is expressed as

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + b_m h(\mathbf{x}; \mathbf{a}). \quad (6)$$

In our system, we implement the MART-based classifier using the library of scikit-learn [39]. Specifically, we choose the deviance function as the loss function, and the learning rate is set to 0.1. Since the MART-based classifier is fairly robust to over-fitting, we set the number of iterations to 5000 to achieve better performance. For each regression tree, the maximal depth is set to 4, and the number of features to consider when looking for the best split is set to 4.

VII. Evaluation

A. Experimental prototype

Since commercial PIN pads or keyboards are not equipped with an ambient light sensor, we built a prototype to mimic the layout and structure of PIN pads that are widely used on ATM machines. As shown in Fig. 8, our prototype consists of five components: a prototype PIN pad (made by cardboard), an LDR-based ambient light sensor (about $1), an analog-to-digital converter (ADS1115 16 bit and 4-channel analog-to-digital converter), a data sink and processing center (Raspberry Pi 3 b+), and a light source (WORKRITE ERGONOMIC VERANO LED array). Since the Raspberry Pi only accepts digital signal from GIPO input, we used a 16-bit converter to convert the analog output to digital signals. On the Raspberry Pi board, we used a Python script and public library to read the sensor data with a frequency of 100 Hz. The LDR-based light sensor is attached in the middle of the PIN pad that is placed under the light source. We implemented our prototype in a shared office room where different human activities exist.

B. Data collection

Our experiments included 10 participants (4 males and 6 females) aged from 22 to 29. All participants are university students who have no knowledge of our system details. We asked each participant who acts as the normal user to randomly choose a 6-digit PIN and input it on our prototype in a comfortable way 43 times. Among them, three instances are used as the template for calculating DTW distances and 20 randomly picked instances are used as training data to build the MART-based classifier. The input behavior of each normal user was recorded by a camera, and some details (e.g. number of fingers used) of normal users' input behaviors are shown in Table I. Additionally, for each normal user, we asked three other participants to act as an attacker. During a simple PIN replay attack, we only gave each of the three attackers the PIN of the victim. Each attacker input the victim's PIN on our prototype in its preferred way 10 times, so we have 30 instances for the simple PIN replay attack. During a strong PIN replay attack, we showed each of the three attackers not only the victim's PIN but also the videos of the victim's input behavior. When each attacker was confident enough to mimic the victim's behavior, the strong PIN replay attack was launched 10 times. For each attacker in both simple and strong PIN replay attacks, five randomly picked instances are used as training data and the remaining five instances are used for testing. Therefore, the training dataset of each user has 20 instances from the normal user, 15 instances from each simple PIN replay attacker, and 15 instances from each strong PIN replay attacker.

To evaluate the performance of our system, we used three metrics, including true acceptance rate (TAR), true rejection rate (TRR), and authentication time. The true acceptance rate is defined as the rate at which a normal user is successfully accepted by the system. Similarly, the true rejection rate is defined as the rate at which an attacker is successfully rejected. The authentication time is defined as the number of PIN input attempts needed to pass our system.

C. System performance for normal users

We first evaluated the system performance for normal users by repeating the experiment 20 times with randomly picked training data and testing data. Fig. 9 shows the average true acceptance rate for 10 participants. We can observe that our system successfully accepts a normal user with an average true acceptance rate of 95%. For user 5, 6, and 8, the average true acceptance rate can reach near 100%. We further study why user 10 has a lower true acceptance rate than other users. We found that the user 10 used the most complex input behavior with 4 fingers in our experiments, which makes her input behaviors less consistent than those of other users and leads to lower true acceptance rate. However, even in the worst case (user 10), our system can still accept the normal user with an average accuracy of at least 89%.

D. System performance against two types of PIN replay attack

With attackers' training data. Similarly, we used the same classifier in Section VII-C and repeated the experiment. The experimental results are illustrated in Fig. 9. It is clear that our system can provide high true rejection rates of about 98% and 96% for both types of PIN replay attacks, respectively. Especially for users 2 and 6, our system can reject all attackers with nearly no errors. We also found that the system performance can decrease to 91% against attackers when the input behavior of a normal user is simple and easy to mimic. For example, user 7 only used his index fingers to input the PIN while

TABLE I
THE PIN INPUT DETAILS OF 10 VOLUNTEERS

User ID	1	2	3	4	5	6	7	8	9	10
Gender	Male	Male	Male	Female	Female	Female	Male	Female	Female	Female
PIN	147536	125836	146928	145832	199423	891218	443659	921218	950131	462856
No. of fingers	1	1	2	1	1	3	1	1	1	4

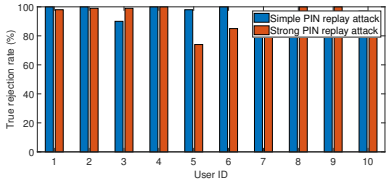


Fig. 10. The system performance without attackers' training data.

other fingers are holding up, which makes it easier for strong attackers to produce similar patterns of received light intensity. Moreover, our system can achieve similar performance for 4-digit PINs with a mean true acceptance rate of about 98.7%, and both types of attackers can be rejected with an accuracy of nearly 100%.

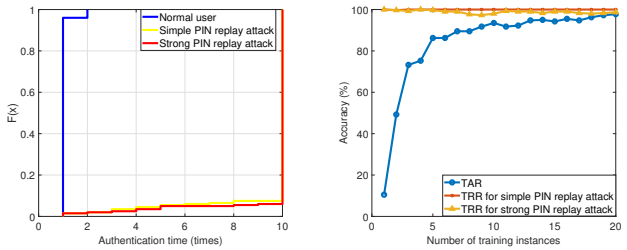
Without attackers' training data. We also evaluated the system performance against attackers whose data is not in our training dataset, which is more common in practice. Fig. 10 illustrates the true rejection rates against two types of attackers. We can see that our system can still ensure high-security protection for users against simple PIN replay attackers with a mean true rejection rate of 96.8%. Moreover, even if strong attackers can imitate the victim's input behaviors, our system can still reject them with mean accuracy of 93.6%

E. Authentication time

The system performance above is for a single PIN input attempt. In practice, to achieve good usability, PIN-based authentication systems usually allow the user to input its PIN for up to three or five times. Therefore, we studied the system performance within the maximum number of input attempts. Fig. 11(a) shows the authentication time distribution of the normal user and two types of PIN replay attackers. If the attacker cannot break our system within ten attempts, its authentication time is set to 10 times to avoid an infinite number. We can see that all normal users can be correctly accepted by our system within two input attempts, while any type of PIN replay attacker is falsely accepted with a possibility of no more than 2%. Even if the attacker can launch the PIN replay attack at most five times, our system can still provide a high true rejection rate of at least 94.5%.

F. Influence of the size of training dataset

In practice, we need to control the size of the training dataset to reduce the cost in the training phase. Therefore, we further studied what is the minimal size of the training dataset needed from the normal user. In our system, we assume that we can collect the attacker's training data in advance for any possible PIN. In this experiment, we randomly selected a normal user and adjusted the training dataset size from 1 to 20 while the training dataset size of two types of attackers was fixed to 30



(a) Impact of authentication time. (b) Impact of training dataset size.
Fig. 11. Authentication time and the influence of training dataset size.

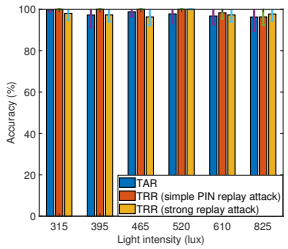
instances. To eliminate the influence of extremely imbalanced training data, we made the normal user's training dataset size constant at 20 by randomly duplicating the training instances. Fig. 11(b) shows the average true acceptance rate and true rejection rate against two types of PIN replay attacks. We can see that true acceptance rate rises with more training instances from the normal user, while the system performance against two types of PIN replay attacks is relatively stable (over 99.7% and 97.3%, respectively) no matter the amount of training instances from the normal user. Specifically, with 9 normal user's training instances, our system can already provide an average true acceptance rate of 91.75%.

G. Influence of light conditions

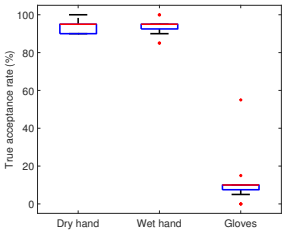
To further evaluate the system performance under lower light intensities, we used an ANNT LED Desk Lamp as the new light source that contains an LED array and can emit lights of five levels from 315 lux to 610 lux. Fig. 12(a) shows the true acceptance rates and true rejection rates under six different light intensities. We can see that the system performance is not influenced by the light intensity of the light source within the range (315 lux to 825 lux) we considered. When the received illuminance is only 315 lux, our system can still correctly accept the normal user and reject the strong PIN replay attacker with an accuracy of at least 98%. Based on our experiment, the average received illuminance is about 350 lux under the fluorescent lamp. Therefore, the light intensity required in our system is comfortable and acceptable for users.

H. Influence of gloves and wet hands

In our default settings, we assume users always interact with our system using dry hands. However, in practice, users may use our system in various conditions, e.g. wet hands in the summer. To evaluate the robustness of our system against various hand conditions, we asked a participant to input his PIN when his hand is wet and in purple nitrile gloves, respectively, and used the classifier that is trained using dry hands to make prediction. Fig. 12(b) shows the evaluation results. We find that our system can still correctly



(a) Different light intensities.



(b) Different hand conditions.

Fig. 12. Influences of different light intensities and hand conditions.

accept the user who used the wet hands with an average true acceptance rate of 93.75%. However, the true acceptance rate drops to about 10% if the user inputs its PIN while wearing gloves. By checking raw output voltage signals, we found that although the overall shapes of output voltage signals are still consistent, the gloves result in higher received light intensity than dry hands. In other words, the gloves change the biological features of the user, which makes the original classifier wrongly detect the user as a strong attacker.

I. Influence of sampling rate

As we discussed in Section V-A, the influences of PIN input on output voltage signals are mainly in the frequency bands from 0 Hz to 2.5 Hz, which means a sampling rate of 5 Hz is enough to capture the information of PIN input in theory. Although we use a high sampling rate of 100 Hz to capture as much information as possible, it is always good to reduce the sampling rate for saving energy. In this experiment, we evaluated the system performance under different sampling rates, and the results are shown in Fig. 13. It is clear that the system performance, especially the true acceptance rate, is improved with the greater sampling rate. When the sampling rate is 5 Hz, the obtained information is enough for our system to provide a high true acceptance rate of 90.5%. By including the high-frequency features, we can achieve an average true acceptance rate of 98% with a sampling rate of 100 Hz.

VIII. Discussion

Long-term stability of input behavior. In our experiments, we already show that the input behavior of the same user is stable within a short term (1 week) for all 10 participants, so we further study the long-term stability of their input behaviors. We invited three participants to input their PIN five weeks after they enrolled in our system. We first use the classifier trained at the beginning to classify their new input and the results that two of them can still be correctly accepted by our system with a true acceptance rate of at least 98%. The input behavior of the remaining one is slightly changed, which introduces variation in the output voltage signals. This problem can be solved by periodically including new instances that represent the behavior changes into the training dataset. Our experiments show that, by adding only five new instances, our system can correctly accept the user with accuracy of 88.5%, while the true rejection rate against strong PIN replay attacker only decreases by 2%.

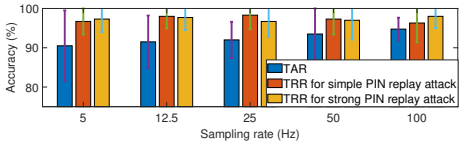


Fig. 13. Influence of sampling rates.

Light sensor in PIN-based authentication systems. For traditional keyboard-based PIN pads, the ambient light sensor can be embedded in the keycap thanks to the small size of the light sensor. For touchscreen-based PIN pads, new techniques enable us to install the light sensor behind an organic light-emitting diode (OLED) display. For example, Austrian Apple supplier AMS announces that they have developed unique algorithms which enable accurate detection of ambient light levels without knowledge of the display pixel brightness above the sensor [40]. Moreover, many wearable devices (e.g. Samsung gear 3 smartwatches) have adopted this new technique. Therefore, we can expect that all PIN pads can have at least one ambient light sensor with reasonable cost.

Limitations and future work. Currently, we only consider input behaviors for people who are of young age and can finish the PIN input quickly. In the future, we also plan to study how slow input behaviors of elder people and the dust influence the system performance. In addition, although our system does not change the original PIN input methods, users still need to provide sufficient training data (at least 10 instances) to obtain acceptable system performance. Besides, our results serve as a feasibility assessment of using ambient light to secure PIN input since our experiments are based on a limited dataset collected from 10 participants. In the future, we will evaluate our approaches on a larger dataset with participants from diverse backgrounds. Moreover, we will further study the influences of dynamic ambient light (e.g. sunlight) and diverse sizes of PIN pads and design new processing techniques to remove their influences.

IX. Conclusion

In this paper, we propose a new system called LightDefender to defend against two types of PIN replay attacks by leveraging the biometrics in the received light intensity that is influenced by input procedure. The key insight is that different input behaviors and biological differences result in different output voltage signals. These differences can be reused as biometrics to authenticate users right after the input procedure. Different from existing approaches, LightDefender does not change the original interaction methods between the user and PIN-based authentication systems, and the extra hardware cost is low. We built a prototype of our system and evaluated it with 10 volunteers. Experimental results show that LightDefender can achieve an average true acceptance rate of 95% for normal users and correctly reject two types of PIN replay attacker with average true rejection rates of at least 93.6%.

Acknowledgement

This research was supported in part by NSF grants CNS 1824440, CNS 1828363, CNS 1757533, CNS 1629746, CNS 1651947, and CNS 1564128.

REFERENCES

[1] N. Berry, https://www.theguardian.com/money/blog/2012/sep/28/debit-cards-currentaccounts, 2012.

[2] D. Shukla and V. Phoha, "Stealing passwords by observing hands movement," *IEEE Transactions on Information Forensics and Security*, 2019.

[3] D. Shukla, R. Kumar, A. Serwadda, and V. V. Phoha, "Beware, your hands reveal your secrets!" in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014, pp. 904–917.

[4] K. S. Balagani, M. Conti, P. Gasti, M. Georgiev, T. Gurtler, D. Lain, C. Miller, K. Molas, N. Samarin, E. Saraci *et al.*, "Silk-tv: Secret information leakage from keystroke timing videos," in *European Symposium on Research in Computer Security*. Springer, 2018, pp. 263–280.

[5] R. Raguram, A. M. White, D. Goswami, F. Monrose, and J.-M. Frahm, "ispy: automatic reconstruction of typed input from compromising reflections," in *Proceedings of the 18th ACM conference on Computer and communications security*. ACM, 2011, pp. 527–536.

[6] L. Lu, J. Yu, Y. Chen, Y. Zhu, X. Xu, G. Xue, and M. Li, "Keylistener: Inferring keystrokes on qwerty keyboard of touch screen through acoustic signals," in *Proceedings of IEEE International Conference on Computer Communications*. IEEE, 2019, pp. 67–78.

[7] C. Wang, X. Guo, Y. Wang, Y. Chen, and B. Liu, "Friend or foe?: Your wearable devices reveal your personal pin," in *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*. ACM, 2016, pp. 189–200.

[8] A. J. Aviv, K. L. Gibson, E. Mossop, M. Blaze, and J. M. Smith, "Smudge attacks on smartphone touch screens." *Woot*, vol. 10, pp. 1–7, 2010.

[9] Z. Xu, K. Bai, and S. Zhu, "Taplogger: Inferring user inputs on smartphone touchscreens using on-board motion sensors," in *Proceedings of the fifth ACM conference on Security and Privacy in Wireless and Mobile Networks*. ACM, 2012, pp. 113–124.

[10] L. Simon and R. Anderson, "Pin skimmer: Inferring pins through the camera and microphone," in *Proceedings of the Third ACM workshop on Security and privacy in smartphones & mobile devices*. ACM, 2013, pp. 67–78.

[11] M.-K. Lee, "Security notions and advanced method for human shoulder-surfing resistant pin-entry," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 4, pp. 695–708, 2014.

[12] A. De Luca, K. Hertzschuch, and H. Hussmann, "Colorpin: securing pin entry through indirect input," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2010, pp. 1103–1106.

[13] V. Roth, K. Richter, and R. Freidinger, "A pin-entry method resilient against shoulder surfing," in *Proceedings of the 11th ACM conference on Computer and communications security*. ACM, 2004, pp. 236–245.

[14] A. De Luca, M. Harbach, E. von Zezschwitz, M.-E. Maurer, B. E. Slawik, H. Hussmann, and M. Smith, "Now you see me, now you don't: protecting smartphone authentication from shoulder surfers," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2014, pp. 2937–2946.

[15] A. De Luca, E. Von Zezschwitz, N. D. H. Nguyen, M.-E. Maurer, E. Rubegni, M. P. Scipioni, and M. Langheinrich, "Back-of-device authentication on smartphones," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2013, pp. 2389–2398.

[16] C. Winkler, J. Gugenheimer, A. De Luca, G. Haas, P. Speidel, D. Dobbelstein, and E. Rukzio, "Glass unlock: Enhancing security of smartphone unlocking through leveraging a private near-eye display," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2015, pp. 1407–1410.

[17] H. Sasamoto, N. Christin, and E. Hayashi, "Undercover: authentication usable in front of prying eyes," in *CHI*, vol. 8, 2008, pp. 183–192.

[18] Q. Yan, J. Han, Y. Li, and R. H. Deng, "On limitations of designing leakage-resilient password systems: Attacks, principles and usability," 2012.

[19] T. Perković, M. Čagalj, and N. Rakić, "Sssl: shoulder surfing safe login," 2010.

[20] A. Bianchi, I. Oakley, and D. S. Kwon, "Spinlock: a single-cue haptic and audio pin input technique for authentication," in *International Workshop on Haptic and Audio Interaction Design*. Springer, 2011, pp. 81–90.

[21] A. Bianchi, I. Oakley, J. K. Lee, and D. S. Kwon, "The haptic wheel: design & evaluation of a tactile password system," in *CHI'10 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2010, pp. 3625–3630.

[22] A. Bianchi, I. Oakley, V. Kostakos, and D. S. Kwon, "The phone lock: audio and haptic shoulder-surfing resistant pin entry methods for mobile devices," in *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction*. ACM, 2011, pp. 197–200.

[23] A. Bianchi, I. Oakley, and D. S. Kwon, "Counting clicks and beeps: Exploring numerosity based haptic and audio pin entry," *Interacting with computers*, vol. 24, no. 5, pp. 409–422, 2012.

[24] A. De Luca, E. Von Zezschwitz, and H. Hußmann, "Vibrapass: secure authentication based on shared lies," in *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 2009, pp. 913–916.

[25] P. Lantz, B. Johansson, M. Hell, and B. Smeets, "Visual cryptography and obfuscation: A use-case for decrypting and deobfuscating information using augmented reality," in *International Conference on Financial Cryptography and Data Security*. Springer, 2015, pp. 261–273.

[26] D. K. Yadav, B. Ionascu, S. V. K. Ongole, A. Roy, and N. Memon, "Design and analysis of shoulder surfing resistant pin based authentication mechanisms on google glass," in *International conference on financial cryptography and data security*. Springer, 2015, pp. 281–297.

[27] J. Thorpe, P. C. Van Oorschot, and A. Somayaji, "Pass-thoughts: authenticating with our minds," in *Proceedings of the 2005 workshop on New security paradigms*. ACM, 2005, pp. 45–56.

[28] M. Kumar, T. Garfinkel, D. Boneh, and T. Winograd, "Reducing shoulder-surfing by using gaze-based password entry," in *Proceedings of the 3rd symposium on Usable privacy and security*. ACM, 2007, pp. 13–19.

[29] A. Salem and M. S. Obaidat, "A novel security scheme for behavioral authentication systems based on keystroke dynamics," *Security and Privacy*, p. e64, 2019.

[30] H. Khan, U. Hengartner, and D. Vogel, "Targeted mimicry attacks on touch input based implicit authentication schemes," in *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 2016, pp. 387–398.

[31] A. De Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann, "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*. ACM, 2012, pp. 987–996.

[32] H. Khan, A. Atwater, and U. Hengartner, "Itus: an implicit authentication framework for android," in *Proceedings of the 20th annual international conference on Mobile computing and networking*. ACM, 2014, pp. 507–518.

[33] M. Shahzad, A. X. Liu, and A. Samuel, "Secure unlocking of mobile touch screen devices by simple gestures: you can see it but you can not do it," in *Proceedings of the 19th annual international conference on Mobile computing & networking*. ACM, 2013, pp. 39–50.

[34] R. H. Venkatnarayan and M. Shahzad, "Gesture recognition using ambient light," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 1, pp. 1–28, 2018.

[35] T. Li, C. An, Z. Tian, A. T. Campbell, and X. Zhou, "Human sensing using visible light communication," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, 2015, pp. 331–344.

[36] T. Li, Q. Liu, and X. Zhou, "Practical human sensing in the light," in *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, 2016, pp. 71–84.

[37] J. Shang and J. Wu, "A usable authentication system usingwrist-worn photoplethysmography sensors on smartwatches," in *Proc. of the IEEE Conference on Communications and Network Security (CNS 2019)*, 2019.

[38] J. H. Friedman and J. J. Meulman, "Multiple additive regression trees with application in epidemiology," *Statistics in medicine*, vol. 22, no. 9, pp. 1365–1381, 2003.

[39] scikit learn. [Online]. Available: https://scikit-learn.org/stable/

[40] C. Hauk, 2019. [Online]. Available: https://www.mactrast.com/2019/01/apple-supplier-announcement-of-new-under-display-light-and-proximity- sensors-means-new-iphone-could-boast-smaller-screen-notch/