An Improved PIN Input Method for the Visually Impaired

N. Caporusso*
* Department of Computer Science, Northern Kentucky University,
Louie B Nunn Dr, 41099 Highland Heights, United States
caporusson1@nku.edu

Abstract – **Despite the recent introduction of biometric identification technology, Personal Identification Numbers (PIN) are the standard for granting access to restricted areas and for authorizing operations on most systems, including mobile phones, payment devices, smart locks. Unfortunately, PINs have several inherent vulnerabilities and expose users to different types of social engineering attacks. Specifically, the risk of shoulder surfing in PIN-based authentication is especially high for individuals who are blind.**

In this paper, we introduce a new method for improving the trade-off between security and accessibility in PIN-based authentication systems. Our proposed solution aims at minimizing the threats posed by malicious agents while maintaining a low level of complexity for the user. We present the method and discuss the results of an evaluation study that demonstrates the advantages of our solution compared to state-of-the-art systems.

Keywords - accessibility; cybersecurity; Human-Computer Interaction

I. INTRODUCTION

In recent years, biometric identification (e.g., using fingerprint, iris, and face recognition) has been introduced as a convenient authentication method for unlocking access to personal technology (e.g., smartphones and smart locks) and restricted areas in offices and buildings. Nevertheless, passcodes based on a Personal Identification Number (PIN) are primarily utilized as the preferred system for authorization purposes in several types of hardware devices and software systems. Indeed, the most common uses are Point-of-Sale payment solutions, which require users to input their PIN to validate their purchase with credit or debit card in Automated Teller Machines (ATMs), payment kiosks, and other types of vending solutions that are available in public locations. Also, most access control systems require users to enter a PIN to grant permission to enter an area or to obtain a resource.

Several studies have shown that, despite their convenience and ease of implementation, PIN-based authentication methods have inherent vulnerabilities and are susceptible to several types of attacks, including those based on brute-force techniques, due to their limited length and entropy [1]. Moreover, social engineering tactics such as shoulder surfing can be utilized to capture user's PIN when they are entering their code on an authentication or authorization device. This can be realized by a nearby attacker or using technology (e.g., hidden cameras or microphones) [2] [3].

In addition to inherent vulnerabilities, the security of current PIN input methods lacks accessibility to users with sensory conditions. Specifically, individuals who are blind are at increased risk of shoulder surfing attacks because they may be unaware of the presence of a nearby malicious agent or device. Although recent research in accessibility introduced new methods for improving the trade-off between security and accessibility, most solutions presented in recent years have important residual vulnerabilities, introduce elements that significantly affect their convenience, or have poor feasibility.

In this paper, we propose a novel accessible input method that is designed to enhance the security of the PIN acquisition process. We detail the rationale of our solution and discuss its implementation on currently available devices based on numeric keypads and on touchscreens and other interfaces that support gestures. Furthermore, we present an evaluation study that confirms that the proposed method can provide users who have vision conditions with an accessible and secure while maintaining a balance in terms of complexity trade-off. Finally, we analyze the drawbacks of our system and potential improvements.

II. RELATED WORK

In the last decades, the level of security of authentication systems based on PINs has been investigated extensively. As the limited entropy of PINs inherently exposes them to brute-force attacks, several systems have been introduced to limit the number of failed attempts. Moreover, research groups have highlighted the vulnerability of PINs to techniques, including shoulder surfing, that enable a malicious agent to retrieve the code by simply looking over the shoulder of their victim as they type the PIN. This can be realized either in person or using cameras and other types of acquisition devices that are becoming increasingly subtle. Recent studies demonstrated how microphones can be utilized as a more elusive yet effective technology for capturing keypresses in PIN input on different systems, including smartphones [2] [3].

As malicious strategies have evolved over the years, more sophisticated PIN input methods have been proposed. However, most of them lack accessibility to individuals who are blind. For instance, [4] proposed the use of gaze detection instead of requiring users to enter their code using

a keypad. Conversely, other solutions especially designed to increase the trade-off between security and accessibility have significant drawbacks. The authors of [5] proposed a novel input method for individuals with vision impairments. Their system associates each of the 10 digits of the PIN with three different words representing colors, body parts, and fruits. Then, in the verification phase, the system presents several objects to the user, who responds to each option with a gesture on a touchscreen that enables marking the symbol as being part of the PIN or as incorrect. By doing this, they increase the entropy in the input method and, thus, make it more difficult for an attacker to use shoulder surfing techniques. Unfortunately, the solution introduces additional complexity for the user, and it is not robust against malicious agents that gain control of the input and output. Other methods focusing on accessibility do not guarantee an adequate level of security. Among them, the PIN input technique especially designed for the blind presented in [6], which does not resolve the issue of shoulder surfing. Finally, more sophisticated techniques that have been proposed recently are not compatible with the devices currently on the market, and their integration may shift the security issue to a different part of the process. For instance, in [7], the authors propose the use of Brain-Computer Interface as an increased-security input method. By analyzing the cortical electrical activity of the user in response to visual or auditory stimuli that represent different options for each digit, their system enables verifying the PIN without any visible action of the user, which prevents the attacker from accessing the user's input. Other solutions involve the use of dedicated devices [8] [9] that provide users with a more private, convenient, and secure input system.

III. System Design

The proposed system is designed to address the limitations of current accessible PIN input methods with specific regard to their security. Also, the objective of our solution is to adapt to the devices currently deployed without requiring significant changes to their hardware. To this end, the proposed input method can be integrated in existing technology that incorporates an input device such as a keypad (e.g., ATMs, POSs, and vending machines) and an output audio/video system, as well as in other types of interfaces, including touchscreens.

The proposed method is not intended to render PINs more secure. Indeed, the complexity of the PIN itself depends on two factors, that is, the number of symbols in the alphabet and the length of the code. Specifically, it can be calculated as σ^{δ}, where σ represents the number of symbols in the alphabet and δ represents the number of digits in the code. As most PINs consist of 10 symbols and 4-6 digits, they involve a relatively low number of combinations (i.e., 10^4-10^6), which renders them especially vulnerable to brute-force attacks [1].

As PINs are less robust than alphanumeric passwords, because of their length and limited alphabet, most PIN-based systems include additional security measures, which mainly consist in limiting the attempts in case of error. As an example, ATMs and payment systems will lock a card after three failed PINs. Also, more effective practices involve avoiding code reuse. For instance, 6-digit codes are often utilized as one-time passwords (OTP) in two-factor authentication (2FA) systems.

In contrast, we propose an input methodology that increases the security of the acquisition process while maintaining the level of complexity of PIN codes unchanged. Specifically, our objective is to improve the four-fold trade-off between (1) accessibility to individuals who are blind, (2) security against social engineering attacks (i.e., shoulder surfing), (3) convenience for the user (i.e., cognitive effort, including ease of learning and use), and (4) feasibility of their implementation in currently available systems and devices.

A. PIN setup

The proposed solution involves adding one extra step to the PIN setup, which consists in defining a confirmation key. Traditional PIN input methods acquire the PIN from the user, who enters the code using a keypad or another form of input interface. Conversely, our approach utilizes the input device to provide the user with two sets of answers: confirm (i.e., Yes) or deny (i.e., No). One symbol from the alphabet (i.e., a number) represents an affirmative answer, whereas the remaining characters are associated with a negative reply. By doing this, by pressing one of the buttons of the numeric keypad, the user will submit either an affirmative or a negative response. We assume that the association between the confirmation code and the symbol of the alphabet is user specific. For instance, it can be defined and associated with the account a priori, so that authentication devices can recognize it: when the user is initially provided with the PIN, they can also select (or be given) the symbol that corresponds to the affirmative response. This association is explained in Figure 1: the number 5 is utilized as a confirmation button, whereas all the others are utilized to submit a negative answer.

B. PIN verification

As in other accessible PIN input systems designed for people with vision and hearing conditions, our solution is based on bi-directional communication between the computer and the human agent. Depending on the characteristics of the user, this can be realized with different types of output techniques, including text-to-speech technology or touch-based displays.

In our method, each digit of the PIN is verified as follows. The system generates a random sequence of two or more symbols from the alphabet (e.g., a series of numbers from 0 to 9) in which at least one character corresponds to the target digit. After presenting one symbol in the sequence, the system waits for a user response before showing the next one: the user can confirm that the symbol is the correct digit or mark it as incorrect (e.g., by pressing either the confirmation code or any other button on the keypad). After the entire sequence has been presented to the user and a response has been received for each symbol, the system moves to the next digit and repeats the process with a new sequence of options. The process terminates after the system has collected the user's input in response to each of the options presented for the digits.

The number of options presented during the verification phase can be configured by the user to obtain a more secure

authentication method at the expense of longer verification time.

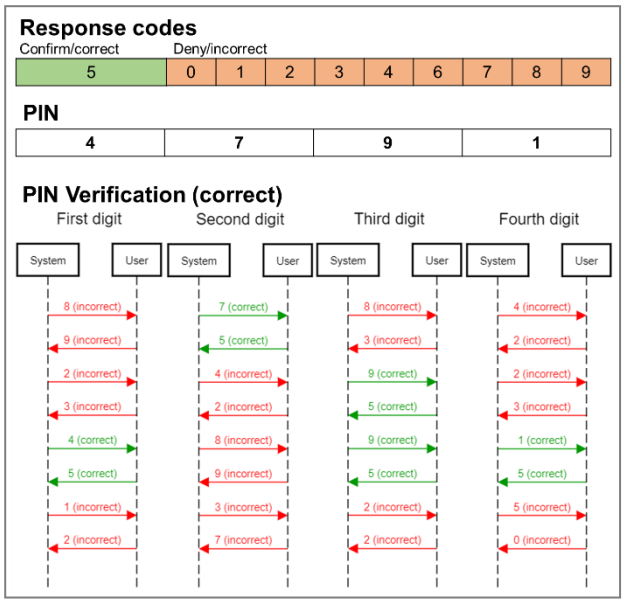

Figure 1. Example of application of the proposed system. In this case, the number 5 enables the user to confirm the correct digit whereas all other numbers are utilized as a negative response. During the verification phase, the system controls the PIN by displaying four options for each digit. At least three are selected at random, whereas one contains the correct digit.

Figure 1 shows an example of a verification process that uses a sequence of four numbers for each digit of the PIN: if the system outputs the correct symbol (e.g., the number 4 as the first digit), the user should press the confirmation button (i.e., 5), whereas in case the symbol is not correct, the user should give a negative answer by pressing any other button. In the example shown in Figure 1, the user should press the button 5 when presented with the number 4 as the first digit, with the number 7 as the second digit, with the number 9 as the third symbol, and with the number 1 as the fourth digit. In all the other cases, the user should respond by pressing a different button. As demonstrated in the sequence diagram, the system starts verifying the PIN by presenting the number 8 as the first option for the first digit. As this is incorrect, the user should respond negatively by pressing any buttons. In this case, the user selects 9. Then, the system presents the second option, that is, the number 2, which is also incorrect. As a result, the user denies that this is the target digit by pressing 3. Subsequently, the system shows the correct symbol, that is, the number 4, as an option. Thus, the user confirms that this is correct by pressing the button associated with the confirmation code (i.e., the number 5). Finally, the system presents the fourth option, that is, 1. As this is incorrect, the user responds negatively by pressing the number 2. After completing the verification of the first digit, the system moves to the second one, which follows the same logic. By doing this, the PIN is verified after the user provides a correct response to each of the sixteen options (that is, four for each digit). As shown in Figure 1, the system can present the same symbol twice (either correct or incorrect), which does not affect the execution of the proposed method, though it has implications in terms of security.

The proposed method completely decouples the input provided by the user and their PIN, which enhances security. Furthermore, the output is displayed via a separated channel, which enables using a more private system (e.g., listening to it using earphones). This, in turn, eliminates the risk of shoulder surfing and smudge attacks.

Indeed, a malicious agent listening to the user's input will be able to identify the response code because the number associated with a positive answer will appear at least once in the verification step of every digit. However, it will be difficult for attackers to exploit the confirmation code if they do not have access to the list options displayed by the device. Specifically, a brute-force attack based on knowledge of the correct confirmation code entered by the user has a likelihood of success equal to the combined probability of a right guess for all the digits, which depends on the number of options. For instance, in the example shown in Figure 1, this is 0.25^4 (i.e., 0.3%), whereas introducing 10 options reduces the probability of success to 0.25^{10}, though it impacts the usability of the method.

Although the proposed method, combined with measures already utilized to prevent brute-force attacks, is robust against shoulder surfing, it is vulnerable to other types of attacks in which a malicious agent gains access to both the system output and the corresponding user input. This can be realized with a combination of social engineering tactics and other hacking techniques. Nevertheless, the proposed input method supports several possibilities for mitigating this risk. One strategy relies on the user and does not involve any changes to the design and implementation of the system. The user can consistently reiterate the same number associated with a negative response to mark numbers displayed by the system as incorrect in the verification step of each digit. As a consequence, negative response codes will appear in the input sequence as many times as the confirmation code. Figure 1 demonstrates this method: in addition to entering the number 5 to respond to a correct option, the number 2 is utilized in every digit to mark output as incorrect. Consequently, both numbers could be interpreted by a malicious agent as a correct response code. By doing this, the user can reduce the success probability of an attack by 50%. As the possibility of consistently using additional incorrect response codes and, thus, using entropy advantageously, depends on the number of options provided as an output for each digit, the success probability of an attack can be reduced up to 90% by asking the user to submit 10 responses for each digit. Unfortunately, this impacts convenience, as discussed in the next Section.

In contrast to other systems for improving PIN security, which require significant hardware, software, and process modifications, the proposed method is designed to work with the traditional PIN acquisition process on devices that are already in use (e.g., ATM, PoS, and eVending) and it requires only minimal changes to the software. Moreover, the input methodology presented in this paper works best with other interfaces that support gesture-based interaction such as touchscreens, because this also enables decoupling the alphabet of the PIN and the symbols utilized as confirmation code. For instance, strokes can be used in response to the number. Although this does not influence security, it increases the usability of the system because it

removes the Stroop Effect caused by the interference between the numbers of the PIN keypad and their meaning as response codes.

IV. System Evaluation

In this Section, we evaluate the security proposed input method, and we compare it with other solutions. Moreover, we present the results of a user study in which we analyzed the usability of our system.

A. Security considerations

If utilized in combination with techniques that prevent brute-force attacks, the proposed acquisition method is robust against social engineering techniques based on shoulder surfing in which a malicious agent can capture the user's input but does not have access to the system output. the level of security of the proposed method is correlated with the number of available options presented by the system for each digit. Specifically, it is calculated as the combined probability of guessing the correct option for each digit. As a result, increasing the number of options results in a more secure PIN. Depending on the implementation of the method, this value can either be fixed or it can be decided by the user during the PIN configuration phase.

Conversely, if the attacker gains control of both the input and the output, the performance of our system from a security standpoint depends on the entropy in the response codes entered by the user across the entire verification process. Although the latter is limited by the number of options presented by the system as an output, in this case, the level of security solely relies on the ability of the user of entering response codes that can confuse the attacker. Lower security levels are obtained either by using one response code for marking all incorrect options or by maximizing by using a number of response codes greater than the number of available options. By doing this, the correct response will be the symbol that appears only once in every digit verification. Conversely, the best result is obtained when the user selects a number of different response codes that is equal to the count of available options and utilizes the same codes consistently in the verification of every digit. By doing this, each symbol will appear the same number of times and, thus, the attacker will be in a situation in which each of the response codes entered by the user has the same probability of being the correct one. In this case, the probability of success of a brute-force attack in which the malicious agent will attempt to use the possible codes is the inverse of the total response codes entered by the user, and it ranges from 0.5 (in the case of 2 options) to 0.1 (in the case of 10 options). Unfortunately, this is still high, though it can be mitigated by systems that prevent further input in case of failed attempts.

B. Usability study

The objective of our preliminary study was to evaluate several components of the usability of the system, including ease of learning, cognitive effort, and perceived ease of use. To this end, we recruited 31 participants, 12 (39%) were females and 19 (61%) were males. Their age ranges were the following: 18-24 (7 - 23%), 25-34 (8 - 26%), 35-44 (8 - 26%), 45-54 (6 - 19%), and 55-64 (2 - 6%). All of them identified as having a normal sight and were familiar with PINs. We designed three different tasks and implemented them in a web-based application that simulated an input device structured as a keypad. Also, the application included output in the form of a pre-recorded sound for each number. Participants received the link to the web page, which enabled accessing each of the experimental tasks after watching a short instruction video. The web application also served as a data-collection tool.

The purpose of the first task was to evaluate the time required for learning the system. To this end, participants were asked to use the confirmation response code to mark a target number as correct and discard all other symbols as incorrect by using other response codes. At the beginning of each trial, the data collection system displayed a new target number and response code. Then, the system presented a sequence of 5 options selected at random and acquired the user's response. Participants were asked to repeat the task until they completed 5 consequent trials without any error. For each participant's session, we recorded the error rate in each trial and the number of participants who were able to complete the trial without errors. Also, we collected individuals' comments at the end of the task. The results are shown in Figure 2. All subjects were able to correctly type their given PINs after 9 trials, though most of them (74%) did not make any errors after they repeated the task 5 times. At the beginning of each session, the average error rate was 0.45, but our data show that it quickly improved below 20% after 2 attempts. Several participants mentioned that they were disoriented by the fact that the response code was also a number, which is consistent with research done in the context of the Stroop Effect. Indeed, the issue can be solved in input interfaces such as touchscreens, by enabling individuals to use stroke colors or gestures instead of numbers. The confirmation could be associated with its corresponding symbol either during the PIN configuration phase or at the beginning of the input verification process. Moreover, we asked participants to rate their likelihood of adopting the proposed method, and their opinion about whether introducing one more element in addition to the PIN number would discourage them from using the system. They unanimously were in favor, given the security and accessibility benefits.

Secondly, our objective was to evaluate the appropriate trade-off between security and convenience in the proposed input method. Thus, in task two, we analyzed the time required to verify the PIN with a number of options that ranged from 2 to 10. At the beginning of the task, participants were provided with a target 4-digit PIN and with a confirmation code that did not change throughout the task. Then, in each trial, the system asked them to verify their PIN by presenting a sequence of options selected at random (including the confirmation code) and by acquiring the user's response. A total of 18 trials were realized, starting with 2 alternatives for each digit and increasing the available number of options every 2 trials. Users had the possibility of changing their last answer in response to an option by typing a cancel key and entering a new response code. By doing this, we could minimize the impact of errors on the trial time.

Figure 3 shows users' response times in the trials. The values do not incorporate the time required by the system to present the output, which depends on the number of options and on the duration of the audio displayed as an output. Total response times vary from 5-30 seconds in case of 2 options to 10-110 seconds in case of 10 options.

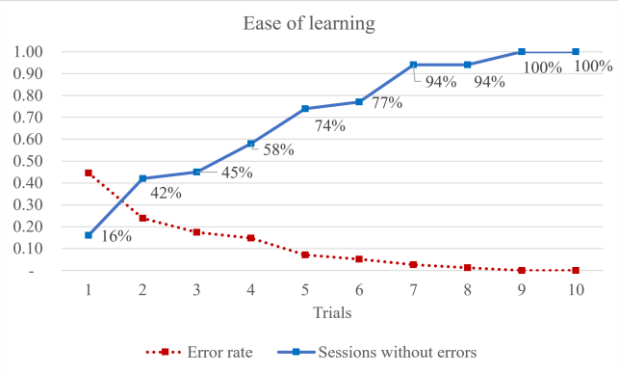

Figure 2. The results obtained in Task 1, which enabled us to measure the ease of learning of the proposed system. To this end, we analized the error rate in each trial (dotted red line) and the number of users who were able to complete the trial without any errors (solid blue line). All users were able to achieve 100% accuracy in at most 9 trials.

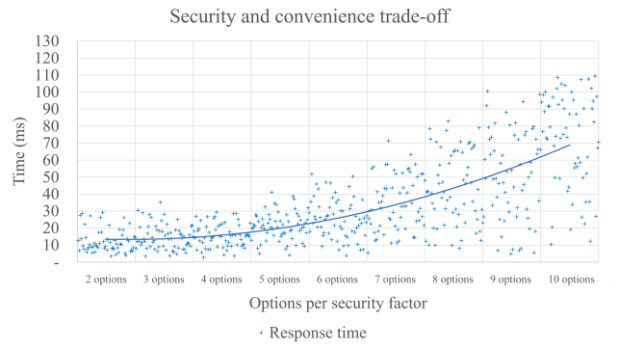

Figure 3. The results obtained in Task 2, in which we investigated the trade-off between security (i.e., the number of available options for each digits) and convenience (i.e., time required to verify the PIN). The chart shows users' response times.

Data from task 2 shows that the proposed method involves a total verification time that is at least one order of magnitude greater than the traditional PIN input. Depending on the number of options, our results show that the total verification process can exceed 2 minutes. Although response times may be impacted by a residual training effect, most users were able to complete the PIN verification phase with significantly less errors compared to the first task. Indeed, using more options results in a more secure verification that is less convenient for the user. Our data show that displaying 10 options corresponds on average to a four-fold increase in the response time and, consequently, in the total time required to complete the task compared to 2-4 options.

TABLE I. PERFORMANCE SUMMARY OF TASK 2

Available options	*System time (ms)*	*Average response time (ms)*	*Average trial time (ms)*	*Weighted response time (ms)*
2	3200	12395 ± 6662	15595	6197
3	4800	14725 ± 7216	19525	4908
4	6400	15517 ± 6828	21917	3879
5	8000	21346 ± 8177	29346	4269
6	9600	25606 ± 11757	35206	4267
7	11200	34842 ± 15813	46042	4977
8	12800	43135 ± 21420	55935	5391
9	14400	51162 ± 25865	65562	5684
10	16000	71836 ± 27556	87836	7183

Table 1 summarizes the average user response time for each of the available options and highlights that the number of available options increases the variance in users' response times. Furthermore, the weighted response time, that is, the average response time for each option, shows that users' performances are consistent across all trials, regardless of the number of options displayed as an output, which demonstrates that subjects were able to learn and use the input method effectively. Nevertheless, the data show better response times in the case of 4-6 options. Although it is counterintuitive that a smaller number of options results in lower performances, this may be because as they are presented with a new digit, they need some time to adjust to it. Conversely, the performance decrease in the case of an increased number of options may be caused by fatigue or other factors associated with attention. Also, we asked users to rate their preference in terms of number of options, and they indicated that they would configure their PIN verification phase to require 3-6. This is in accordance with the data measured from their performances and, specifically, with the weighted response time.

In task 3, we aimed at evaluating whether changing the response code impacts the usability of the proposed method. To this end, participants were asked to verify a total of 8 different PINs. Four of them utilized the same confirmation code, whereas the other half involved a different confirmation code each time. At the beginning of each trial, the user was provided with the PIN and with the confirmation code. Four alternative options were provided. We divided participants in two groups: one began the task with four trials providing different confirmation codes, whereas the other group started with four consecutive trials that had the same confirmation code. As shown in Figure 4, using different confirmation codes results in slightly lower performances. However, this may be due to the cognitive load caused by the experiment design itself, which required the participants to change multiple confirmation codes in a short time.

Finally, task 4 was identical to task 3, with the only modification consisting in using different shapes instead of numeric symbols on the keypad. As shown in Figure 4, this resulted in slightly better performances. Furthermore, users confirmed that they preferred this method, mainly because it removed the interference between the confirmation code and the options displayed as an output.

In the last task, we utilized shapes as confirmation codes instead of colors or objects (as realized by many studies in the literature, including [4] and [5]), because they can be easily implemented in touchscreen devices. This enabled us to evaluate the potential integration of our system as an additional PIN input method in novel interfaces. By doing this, they inherently offer an additional

security measure, because users will be able to define their own shape in the PIN configuration phase and utilize other user-defined shapes in response to incorrect options presented by the system.

We tested the system with sighted participants before realizing a study with individuals who are blind, which is part of our future work.

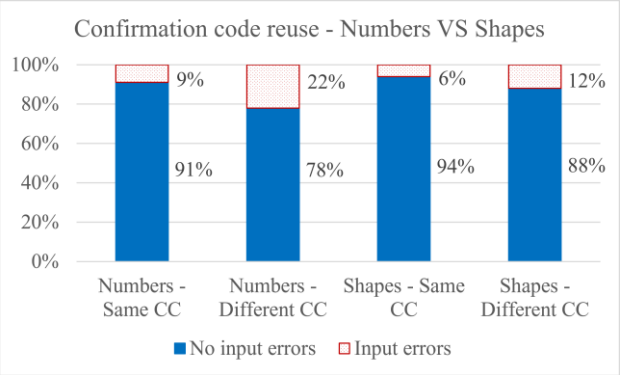

Figure 4. Performance difference in the code reuse. The chart summarizes the data collected in tasks 3 and 4. As shown in the Figure, reusing the same confirmation code results in an accuracy increase in the verification phase of PINs. Also, using shapes instead of numeric confirmation codes increases subjects' performances and results in a better acceptance from individuals who participated in the study.

V. Conclusion

Enabling individuals who are blind to conveniently and securely access technology is an important aspect of universal access and use [10]. Passcodes based on PINs are still the preferred and most common authentication method in many scenarios, including mobile technology, ATMs, POS devices, and access control systems, despite their security vulnerabilities. Recent studies focusing on accessibility highlighted that individuals who are blind may be especially subject to attacks based on social engineering and they proposed novel input techniques that improve security by adding significant complexity.

In this paper, we introduced a PIN input method that ensures accessibility to individuals who are blind while maintaining a low level of complexity compared to state-of-the-art systems and other novel solutions. Our approach consists in defining a user-specific confirmation code that can be utilized during the verification phase of the PIN. In our method, for each digit of the PIN, the system outputs a number of options in sequence. The user selects the correct one using the confirmation code and they mark the others as incorrect by realizing an alternative action in the input interface (e.g., the keypad). By doing this, our method decouples the PIN and the code entered by the user, which is beneficial against shoulder surfing. We presented an example of its use and discussed its advantages and drawbacks from a security standpoint. The proposed system can be integrated in currently available technology as well as in novel devices and interfaces based on touch or leveraging non-conventional input methods [11] [12]. Also, we detailed a user study that enabled us to evaluate the usability of our system with specific regard to user's cognitive effort and we identified a security and convenience trade-off that may be suitable for most users.

References

[1] Wang, D., Gu, Q., Huang, X. and Wang, P., 2017, April. Understanding human-chosen pins: characteristics, distribution and security. In Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security (pp. 372-385).

[2] Panda, S., Liu, Y., Hancke, G.P. and Qureshi, U.M., 2020. Behavioral Acoustic Emanations: Attack and Verification of PIN Entry Using Keypress Sounds. Sensors, 20(11), p.3015.

[3] Cardaioli, M., Conti, M., Balagani, K. and Gasti, P., 2019. Your PIN Sounds Good! On The Feasibility of PIN Inference Through Audio Leakage. arXiv preprint arXiv:1905.08742

[4] AlBaradi, B.M., AlTowayan, A.M., AlAnazi, M.M., Ambreen, S. and Ibrahim, D.M., PathGazePIN: Gaze and Path-based Authentication Entry Method.

[5] Jeon, I.S. and Kim, M.S., 2017. An Enhanced Simple PIN Input Technique Resisting Shoulder Surfing and Smudge Attacks. Contemporary Engineering Sciences, 10(5), pp.203-210.

[6] Kuber, R. and Sharma, S., 2010, October. Toward tactile authentication for blind users. In Proceedings of the 12th international ACM SIGACCESS conference on Computers and accessibility (pp. 289-290).

[7] Saulynas, S., Lechner, C. and Kuber, R., 2018. Towards the use of brain–computer interface and gestural technologies as a potential alternative to PIN authentication. International Journal of Human–Computer Interaction, 34(5), pp.433-444.

[8] Caporusso, N., 2008, May. A wearable Malossi alphabet interface for deafblind people. In Proceedings of the working conference on Advanced visual interfaces (pp. 445-448).

[9] Caporusso, N., Biasi, L., Cinquepalmi, G., Trotta, G.F., Brunetti, A. and Bevilacqua, V., 2017, July. A wearable device supporting multiple touch-and gesture-based languages for the deaf-blind. In International Conference on Applied Human Factors and Ergonomics (pp. 32-41). Springer, Cham.

[10] Caporusso, N., Trizio, M. and Perrone, G., 2014. Pervasive assistive technology for the deaf-blind need, emergency and assistance through the sense of touch. In Pervasive health (pp. 289-316). Springer, London.

[11] Caporusso, N., Biasi, L., Cinquepalmi, G., Trotta, G.F., Brunetti, A. and Bevilacqua, V., 2017, July. Enabling touch-based communication in wearable devices for people with sensory and multisensory impairments. In International Conference on Applied Human Factors and Ergonomics (pp. 149-159). Springer, Cham.

[12] Caporusso, N., Mkrtchyan, L. and Badia, L., 2009. A multimodal interface device for online board games designed for sight-impaired people. IEEE Transactions on Information Technology in Biomedicine, 14(2), pp.248-254.