Augmented Unlocking Techniques for Smartphones Using Pre-Touch Information

Matthew Lakier
matthew.lakier@uwaterloo.ca
University of Waterloo

Dimcho Karakashev
dzkaraka@uwaterloo.ca
University of Waterloo

Yixin Wang
y3244wan@uwaterloo.ca
University of Waterloo

Ian Goldberg
iang@uwaterloo.ca
University of Waterloo

ABSTRACT

Smartphones secure a significant amount of personal and private information, and are playing an increasingly important role in people's lives. However, current techniques to manually authenticate to smartphones have failed in both not-so-surprising (shoulder surfing) and quite surprising (smudge attacks) ways. In this work, we propose a new technique called 3D Pattern. Our 3D Pattern technique takes advantage of pre-touch sensing, which could soon allow smartphones to sense a user's finger position at some distance from the screen. We describe and implement the technique, and evaluate it in a small pilot study (n=6) by comparing it to PIN and pattern locks. Our results show that although our prototype takes longer to authenticate, it is completely immune to smudge attacks and promises to be more resistant to shoulder surfing.

CCS CONCEPTS

• **Security and privacy → Usability in security and privacy**; • **Human-centered computing → Interaction techniques**.

KEYWORDS

authentication, mobile devices, pre-touch

ACM Reference Format:
Matthew Lakier, Dimcho Karakashev, Yixin Wang, and Ian Goldberg. 2020. Augmented Unlocking Techniques for Smartphones Using Pre-Touch Information. In *Symposium on Spatial User Interaction (SUI '20), October 31-November 1, 2020, Virtual Event, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3385959.3418455

Figure 1: Study participant authenticating using our 3D Pattern technique.

1 INTRODUCTION

Smartphones are increasingly used to secure private information such as personal photos, contacts, and financial information. However, smartphones are also frequently used in public spaces or in social gatherings, necessitating the protection of this private information via user authentication. Authentication or "unlocking" techniques include manual (e.g., PINs and gesture-based pattern locks) and biometric (e.g., fingerprint reading, iris scanning, and face recognition) techniques.

In this work, we focus on manual authentication techniques, because they are among the most common techniques used on smartphones as a way to protect private information. Even users utilizing fingerprint readers are often required to enter a PIN for added security, for example, when rebooting or authorizing payments. However, common authentication techniques often have surprising failure modes. We examine in particular the effect of the "smudge attack" [1] whereby swiping an unlock pattern on the screen leaves a readily visible oily smudge that unintentionally reveals some or all of the pattern.

Our position is that we should leverage *pre-touch sensing* technology to create a new authentication technique that, unlike previous solutions, has the user manually authenticate *without touching the screen at all.* Our technique is immune to the smudge attack, and also promises to be less prone to shoulder surfing attacks [3]. *Pre-touch sensing* uses information about a user's fingers just before the screen is actually touched [6, 7]. We create a novel version of the Android pattern lock that expands the traditional 3×3 grid out of the screen into a $3 \times 3 \times 3$ cube. Points are connected by moving a finger in 3D space above the surface of the phone. Because pre-touch information is not available on current smartphones, we simulate it using a motion capture system, enabling a prototype implementation of the 3D Pattern lock. In 2016, Hinckley et al. [6] explored how a smartphone with a self-capacitance touchscreen could enable pre-touch input. We envision that our pre-touch authentication technique could be adapted to use a similar approach, for use on smartphones without additional motion tracking hardware.

2 RELATED WORK

Our 3D Pattern lock is related to conventional 2D pattern locks and locks focusing on small-scale interactions.

De Luca et al. suggested using a stroke-based visual authentication scheme [2], expecting visual patterns to be easier to remember compared to conventional numeric PINs or alphanumeric passwords. A similar technique, the pattern lock, was ultimately

SUI '20, October 31-November 1, 2020, Virtual Event, Canada

ACM ISBN 978-1-4503-7943-4/20/10…$15.00
https://doi.org/10.1145/3385959.3418455

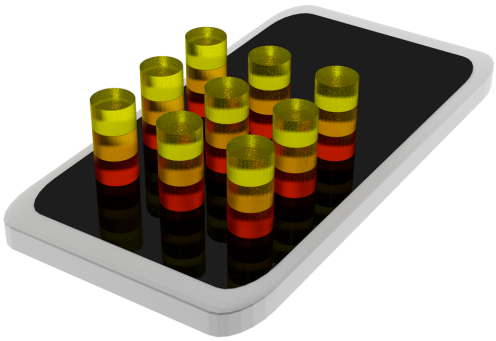
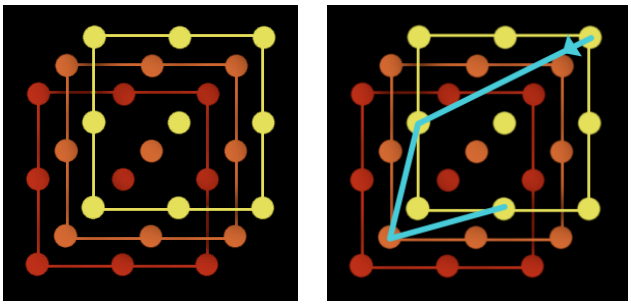


Figure 2: Cylinder representation of our 3D Pattern technique. Colours represent the layer. Users must only hold their finger in the cylinder on a layer to select the corresponding point during authentication.



(a) A screenshot of the smartphone screen as the user sees it before authenticating using the 3D Pattern technique.

(b) An example reference image shown to a participant when authenticating using the 3D Pattern technique.

Figure 3: The on-screen representation of the 3D Pattern technique. The 3D cube is orthogonally projected on the screen, with each layer represented using a different colour.

incorporated into the Android operating system. Unfortunately, as previously mentioned, pattern locks have been shown to be weak against shoulder surfing and smudge attacks. In contrast, DRAW-A-PIN, by Nguyen et al. [9], has users draw each PIN digit on the screen using their finger. Results indicated that this approach could mitigate shoulder surfing attacks.

Several works have attempted to mitigate the impact of smudge attacks by limiting the touch interaction to a small area on the screen. TinyLock [8] is a small version of the Android pattern lock. After authenticating, the user rotates a virtual wheel on top of the grid, distorting the smudges from the pattern. Similarly, ClickPattern [4] shows a keypad in a randomly shuffled order in a small area at the bottom of the screen. Our 3D Pattern technique avoids the need to limit the interaction area by not requiring the user's finger to make contact with the screen.

3 3D PATTERN LOCK

The 3D Pattern lock is inspired by Android's conventional pattern lock. A simple adaptation of the pattern lock to a pre-touch environment would be to duplicate the normal 3×3 grid of points, but have users enter the pattern with their finger hovering over the screen rather than touching it, thus eliminating the smudge attack vector. This simple "hover pattern" use of pre-touch should maintain most of the security and usability properties of the normal pattern lock, save for being immune to smudge attacks, and so we do not analyze this technique in detail. Instead, we extend the pattern lock concept into a full third "z" dimension. Rather than a 3×3 grid of circles, our 3D Pattern lock is a $3 \times 3 \times 3$ cube of cylinders (see Figure 2). The smartphone renders an orthogonal projection of the cube (see Figure 3a). Each depth, or "layer", of the cube is represented in a different colour. The user authenticates by connecting the points in a chosen sequence. A point is selected immediately when the finger enters the corresponding cylinder. Users are not required to slide their fingers on the screen. This inherently protects against smudge attacks since users will not leave oily residues on the screen.

Assuming that the user is allowed to connect any four points such that no point is reused, a theoretical password space upper bound is $27 \times 26 \times 24 \times 23 = 387504$ patterns. However, to improve usability, we limit the space of valid 3D patterns to include only those that start on the topmost layer, do not bypass the middle layer, do not bypass a point within a layer or use a point more than once (as with the traditional Android pattern lock), and do not connect points across a layer with a distance of more than $\sqrt{3}$ units (to avoid difficult-to-input diagonal lines). Using a recursive algorithm in Python, we found 19192 possible four-digit 3D Patterns. This means the password space for our 3D Pattern technique is better than that of both PIN (10000) and pattern (1400, computed using a similar Python script as above) locks for a four-digit PIN or pattern.

As with the conventional Android pattern lock, our 3D Pattern lock has two modes: (1) *with feedback* and (2) *without feedback*. In *with feedback* mode, as the user's finger moves between the different points, a line is rendered between each connected point. In *without feedback* mode, these lines are not rendered.

We implemented the 3D Pattern lock, as well as traditional PIN and (2D) pattern locks for comparison, with additional haptic feedback. In the ambient noise of the experiment room, the experimenters could not hear or otherwise detect the haptic feedback as participants authenticated. Similarly, an attacker should not be able to hear the haptic feedback in a public environment.

Our 3D Pattern lock additionally always renders a "cursor" on the screen. The cursor changes colour depending on the finger's distance from the screen. Yellow represents that the finger is in the closest layer to the user, orange represents the middle layer, and red represents the layer closest to the screen. Figure 1 demonstrates the authentication process.

3.1 Implementation with Motion Capture

Pre-touch information is not yet available on current commercial smartphones. To gain a better understanding of how pre-touch could work for smartphone authentication, we simulated pre-touch capabilities with fiducial-based motion capture. A $2\,\text{m} \times 2\,\text{m} \times 3\,\text{m}$ room was instrumented with six Vicon motion-capture cameras, which track both the smartphone and the finger positions. The absolute positions of these objects in 3D space are transformed, resulting in finger coordinates relative to the phone screen.

Our prototype implementation includes a main PC, which determines what to draw on the phone screen, controls the experiment, verifies PINs, and logs useful information. The three authentication techniques were implemented using the Unity game engine.[1] The source code is available at https://github.com/spamalot/3D-Pattern-Lock. All six Vicon cameras are connected to a server through a network switch. This server calculates the absolute 3D positions of the user's finger and smartphone and forwards this information to the main PC. The connection between the smartphone and the main PC is implemented as a client-server architecture over a Wi-Fi connection. The smartphone renders the authentication technique to the user and accepts touch and dragging input. If our system were to be implemented in practice, of course, all computation and sensing would be performed on the smartphone itself.

To evaluate our technique, we compared it to two other popular authentication techniques: PIN and (regular 2D) pattern locks.

4 PILOT EXPERIMENT

We conducted a small pilot experiment with six participants to understand how quickly and accurately users can authenticate using the 3D Pattern technique, and the resistance of the technique against shoulder surfing. The participants were computer science graduate students (5 male, 1 female) with average age 24 (SD=2). All participants were right-handed, and none had prior experience with the technique. We outline the experiment and results here.

4.1 Task

The experiment had two sections. In the first section, the participant was instructed to authenticate using each of the three techniques. Input events on the smartphone were logged on the main PC and videos of participants authenticating were recorded. All PINs and patterns were randomly generated, and were four "digits" long; that is, PINs had four numbers and patterns involved connecting four points. Participants were allowed to look at the reference PINs and patterns on a separate computer monitor. The pattern and 3D Pattern reference images were rendered as they would be seen after being entered on the phone screen (see Figure 3b).

In the second section, the participant was asked to shoulder surf the videos of the previous participant authenticating *without feedback*; the last successful (most practiced) authentication for each technique was shown. The first participant shoulder surfed once the last participant finished authenticating using all techniques. The participant had up to 20 guesses to correctly determine the PIN or pattern entered. We chose to have our participants, who used our 3D Pattern technique, be the shoulder surfers because they were familiar with this novel technique.

4.2 Design and Procedure

The study was a within-subjects design with TECHNIQUE and TRIAL NUMBER as independent variables. Technique *Entry Time*, technique *Error Rate* (percentage of failed trials), and shoulder surfing *Guesses* were measured as dependent variables. TECHNIQUE had 3 levels: PIN, Pattern, and Pattern3D, the last of which corresponded to our 3D Pattern design.

[1] https://unity3d.com

Techniques were ordered following a Latin square. For each technique, there were two PINs or patterns. For each PIN or pattern, there were two blocks of five authentication trials. Failed trials were not repeated. The first block of each technique was a practice round, and the data were not analyzed. The only difference between the practice block and second block was that the practice block of Pattern and Pattern3D rendered *with feedback*, whereas during the second block, the techniques rendered *without feedback*.

4.3 Results

A repeated measures ANOVA with Greenhouse-Geisser sphericity correction found a significant main effect of TECHNIQUE on log-transformed *Entry Time* ($F_{1.36,6.84} = 22.79, p < 0.01, \eta^2_G = 0.71$). Post hoc paired t-tests with Holm correction show with significance that Pattern3D was slower than PIN ($p < 0.0001$) and Pattern ($p < 0.0001$), and that PIN was slower than Pattern ($p < 0.001$). The median "time from first digit", or the difference in time between the first input towards authenticating and finishing authentication, using Pattern was 1.3 seconds (IQR=0.8), PIN was 2.0 seconds (IQR=1.0), and Pattern3D was 4.7 seconds (IQR=4.4).

A Friedman rank sum test shows a significant effect of TECHNIQUE on *Error Rate* ($\chi^2_3 = 17.72, p < 0.001$). Post hoc paired Wilcoxon signed-rank tests with Holm correction show with significance that Pattern3D has a higher error rate than PIN ($p < 0.0001$) and Pattern ($p < 0.0001$), but do not indicate any significant difference between PIN and Pattern ($p = 0.57$). The mean error rate for PIN was 3% (SD=18%), Pattern was 2% (SD=13%), and Pattern3D was 52% (SD=50%).

An empirical CDF (cumulative distribution function) representing the number of *Guesses* needed to correctly guess a PIN or pattern from a shoulder-surfing video is depicted in Figure 4. PIN and Pattern are similar in shoulder-surfing resistance, whereas Pattern3D appears to have a slight advantage. A Friedman rank sum test shows a significant effect of TECHNIQUE on *Guesses* ($\chi^2_3 = 9.4, p < 0.05$). Post hoc paired Wilcoxon signed-rank tests with Holm correction show with significance that Pattern3D was harder to guess correctly than PIN ($p < 0.01$), but shows no other significant effects. The mean number of guesses needed for PIN was 1.3 (SD=1, median=1), Pattern was 2.0 (SD=1, median=1.5), and Pattern3D was 5.3 (SD=6, median=2.5). One participant was not able to guess one 3D Pattern within the given 20 trials.

5 DISCUSSION

Based on our experience designing and evaluating the 3D Pattern technique, we discuss its shoulder-surfing resistance, and possible future directions for exploration.

Shoulder-Surfing Resistance. Statistical analysis shows that the shoulder-surfing resistance of the 3D Pattern technique is higher than that of PIN locks. The empirical CDF of shoulder-surfing guesses (Figure 4) might indicate that it is also more shoulder surfing resistant than the pattern technique. Because users hover their fingers in 3D space, it is hard for a shoulder surfer to guess the layer in which the user's finger hovers.

Previous work [10, 11] has used picture-based passwords to improve memorability. Our technique could be expanded to combine

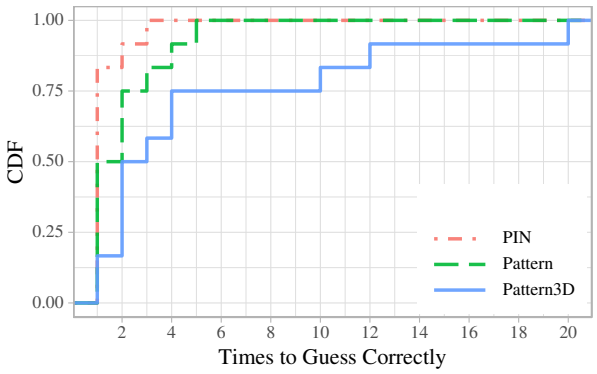


Figure 4: Empirical CDF of guesses needed to correctly identify the PIN or pattern in the shoulder surfing video.

the ideas of both picture-based passwords and pre-touch information. This would allow for the creation of meaningful and easily remembered passwords that are less sensitive to finger positioning.

Equipment Considerations. Motion capture systems can be very accurate when tracking, but this depends on a number of environmental factors such as lighting and camera placement. In our study, there was some visible jitter in the position of the tracked objects, likely slowing down participants when using the 3D Pattern technique. While not reported by participants, the double-sided tape attaching the fiducial marker to the finger may have caused discomfort, increasing authentication time.

Compared to previous studies [5], our implementations of both PIN and pattern authentication were slower. Due to network latency, there was a small amount of cursor lag in all three techniques, meaning reported authentication times may be slightly overestimated. We also found that, during authentication, participants sometimes took extra time when referring back to the secondary computer monitor to recall which PIN or pattern to enter.

Despite the shortcomings of our use of motion capture and the network setup in our study, with several modifications, motion capture has several qualities that make it well suited to investigating pre-touch smartphone authentication techniques. Future studies should use more motion capture cameras, and the network setup should be tailored to reduce latency (e.g., using a lower level networking library than that provided by a game engine). The precise tracking of the finger location provided by motion capture could help develop improved hover position layouts, guided by an analysis of users' finger trajectories. Further, motion capture requires limited additional objects to be affixed to the phone, allowing the user to authenticate while holding the phone with a realistic grip. Given these qualities, a motion capture approach would be useful for estimating a lower bound for how quickly and accurately a user can authenticate with a pre-touch technique, once the hardware for pre-touch sensing becomes available in phones.

Experimental Protocol. Our experiment allowed the participant to practice each technique five times. Given the novelty of pre-touch interfaces, this might not have been enough practice for participants to become accustomed to this new paradigm. Further in support of this argument is the fact that participants took several seconds between starting authentication and entering the first digit for the 3D Pattern technique. Users needed to adjust their finger position to find the top layer.

Apart from the aforementioned equipment considerations and the novelty of pre-touch to participants, authentication times for the 3D Pattern technique may still be inherently longer than traditional screen locks. There is a usability-security tradeoff associated with the design of authentication techniques, and the 3D Pattern technique uses a more complex input approach to achieve a larger password space and potential for increased shoulder-surfing resistance. Further studies on pre-touch input could inform the design of faster pre-touch authentication techniques through an investigation of human pre-touch performance capabilities. Future work could study pre-touch authentication at a larger scale, and also employ qualitative methods to collect feedback about user preferences for different types of authentication techniques.

Visualization of the Cube. The choice of representation for the 3D cube on the 2D phone screen is important for the technique's authentication speed and error rate. One possible extension would be to investigate if perspective projection would result in better performance than our implemented orthogonal projection. It could also be effective to slightly rotate the projection of the cube depending on the finger position or orientation of the smartphone.

Another possible extension would be to use different kinds of depth cues. Our implementation uses a cursor with varying colours and sizes based on depth. This has the notable problem of providing additional information to a shoulder surfer. To improve shoulder-surfing resistance, future work should aim to minimize the number of visual clues. Instead of cursor colours, shadows could be rendered to give an impression of finger height, or a depth-of-field blurring effect based on the finger's distance from the screen could be applied. Alternatively, transitions between finger distances could be indicated, instead of absolute finger distance.

Adding a dwell time to select a point in the authentication screen could potentially increase the accuracy of the 3D Pattern technique. However, the majority of participant errors were not across cylinders in separate layers, but rather along the plane of the screen. Because the spacing of selection points in this plane were matched to that of the Android lock screen, the value of adding dwell time for selection is unclear.

6 CONCLUSION

In this work, we have proposed a novel approach to smartphone authentication using pre-touch information, called 3D Pattern. 3D Pattern is naturally immune to the "smudge attack", which is a surprising failure mode of existing manual authentication techniques. We have implemented a prototype of 3D Pattern by simulating a pre-touch-capable smartphone using a motion capture system. We have also evaluated the 3D Pattern technique in a pilot study, in comparison to two popular existing techniques, finding that authentication times were longer, but that the technique could be more resistant to shoulder-surfing attacks, while being immune to smudge attacks. We attribute the longer authentication times mainly to environmental conditions adversely affecting motion capture and the novelty of pre-touch to participants. We believe these limitations could be mitigated as pre-touch becomes mainstream.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their helpful comments for improving this paper. This work was made possible by NSERC Discovery Grants 2016-03878, 2017-03858, and 2018-05187, the Canada Foundation for Innovation Infrastructure Fund "Facility for Fully Interactive Physio-digital Spaces" (#33151), and Ontario Early Researcher Award #ER16-12-184. This research was undertaken, in part, thanks to funding from the Canada Research Chairs program.

REFERENCES

[1] Adam J. Aviv, Katherine Gibson, Evan Mossop, Matt Blaze, and Jonathan M. Smith. 2010. Smudge Attacks on Smartphone Touch Screens. In *Proceedings of the 4th USENIX Conference on Offensive Technologies* (Washington, DC) *(WOOT'10)*. USENIX Association, Berkeley, CA, USA, 1–7. http://dl.acm.org/citation.cfm?id=1925004.1925009

[2] Alexander De Luca, Roman Weiss, and Heinrich Hussmann. 2007. PassShape: Stroke Based Shape Passwords. In *OZCHI '07* (Adelaide, Australia). ACM, New York, NY, USA, 239–240. https://doi.org/10.1145/1324892.1324943

[3] Malin Eiband, Mohamed Khamis, Emanuel von Zezschwitz, Heinrich Hussmann, and Florian Alt. 2017. Understanding Shoulder Surfing in the Wild: Stories from Users and Observers. In *CHI '17* (Denver, Colorado, USA). ACM, New York, NY, USA, 4254–4265. https://doi.org/10.1145/3025453.3025636

[4] Meriem Guerar, Alessio Merlo, and Mauro Migliardi. 2017. Clickpattern: A pattern lock system resilient to smudge and side-channel attacks. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications* 8 (January 2017), 64–78.

[5] Marian Harbach, Alexander De Luca, and Serge Egelman. 2016. The Anatomy of Smartphone Unlocking: A Field Study of Android Lock Screens. In *CHI '16* (San Jose, California, USA). ACM, New York, NY, USA, 4806–4817. https://doi.org/10.1145/2858036.2858267

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

[7] Insu Kim, Keunwoo Park, Youngwoo Yoon, and Geehyuk Lee. 2018. Touch180: Finger Identification on Mobile Touchscreen Using Fisheye Camera and Convolutional Neural Network. In *UIST '18 Adjunct Proceedings* (Berlin, Germany). ACM, New York, NY, USA, 29–32. https://doi.org/10.1145/3266037.3266091

[8] Taekyoung Kwon and Sarang Na. 2014. TinyLock: Affordable defense against smudge attacks on smartphone pattern lock systems. *Computers and Security* 42 (2014), 137 – 150. https://doi.org/10.1016/j.cose.2013.12.001

[9] Toan Van Nguyen, Napa Sae-Bae, and Nasir Memon. 2017. DRAW-A-PIN: Authentication using finger-drawn PIN on touch devices. *Computers & Security* 66 (2017), 115 – 128. https://doi.org/10.1016/j.cose.2017.01.008

[10] Ilesanmi Olade, Hai-Ning Liang, and Charles Fleming. 2018. SemanticLock: An authentication method for mobile devices using semantically-linked images. *CoRR* abs/1806.11361 (2018).

[11] Tetsuji Takada and Hideki Koike. 2003. Awase-E: Image-Based Authentication for Mobile Phones Using User's Favorite Images. In *Human-Computer Interaction with Mobile Devices and Services*, Luca Chittaro (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 347–351.