Gaze-Based Authentication Method Using Graphical Passwords Featuring Keypoints

Yuki Yamato
yamato@iplab.cs.tsukuba.ac.jp
University of Tsukuba
Tsukuba, Ibaraki, Japan

Shin Takahashi
shin@cs.tsukuba.ac.jp
University of Tsukuba
Tsukuba, Ibaraki, Japan

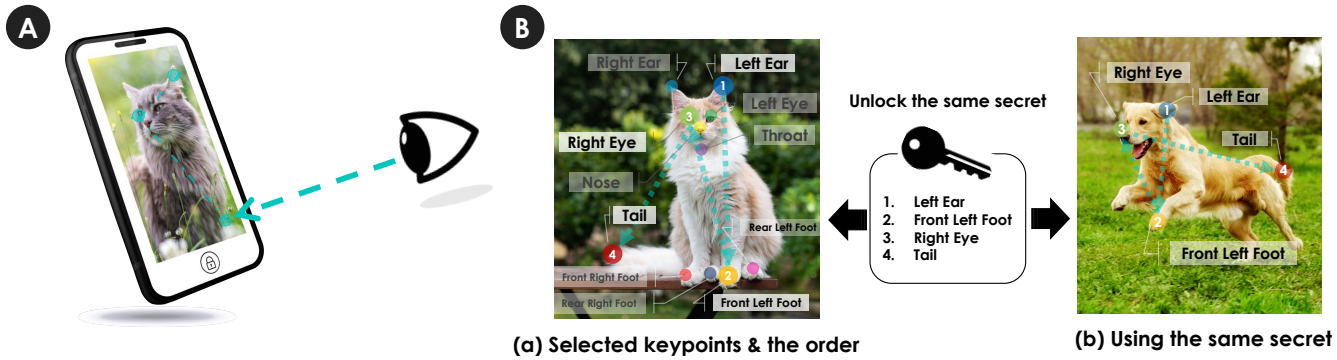

Figure 1: (A) A concept image of our proposed method. The user unlocks the device by gazing at the keypoints in a displayed image in order. (B) The user registers an ordered list of the names of keypoints as a secret. Although the system displays a different image each time, the user can unlock the device with the same secret.

ABSTRACT

We propose a gaze-based authentication method using a new type of graphical password. After registering keypoints in an image as a secret, the user can unlock the device by gazing at them in the registered order on a randomly displayed image. We present the design and implementation of our authentication method and report the results of a feasibility study utilizing mobile devices. Our prototype implementation achieved an acceptance rate of 86.0%.

CCS CONCEPTS

• **Security and privacy** → *Graphical / visual passwords*; *Operating systems security*; • **Human-centered computing** → **Human computer interaction (HCI)**.

KEYWORDS

Gaze-Based Authentication, Graphical Passwords, Keypoints, Gaze Input

ACM Reference Format:
Yuki Yamato and Shin Takahashi. 2021. Gaze-Based Authentication Method Using Graphical Passwords Featuring Keypoints. In *33rd Australian Conference on Human-Computer Interaction (OzCHI '21), November 30-December 2, 2021, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3520495.3520527

OzCHI '21, November 30-December 2, 2021, Melbourne, VIC, Australia

1 INTRODUCTION

There are various authentication methods for personal identification to protect private information. The two mostly widely used are knowledge-based authentication, such as PIN and password, and biometric authentication, such as fingerprint, face, and vein authentication. Knowledge-based authentication is still the mainstream method [10]. As for biometric authentication, while it provides sufficiently strong security, instant verification, and convenience for users [1], once the secret is stolen, it cannot be reset or reissued, posing a lifelong security risk to the user. In contrast, knowledge-based authentication features easily replaceable secrets. However, the authentication method that uses just numbers or characters (e.g., PIN or passwords) is exposed to dictionary attacks and brute force attacks.

Graphical passwords [5, 16, 23] that are robust against such attacks have been proposed. They have the advantages of memorability [16] and are more secure than PINs/passwords [5]. They are also resistant to guessing attacks because their password space is theoretically larger. There are two types of graphical passwords: a pattern of selecting multiple images [5] and a pattern of selecting multiple points in an image [23]. However, using these types of passwords requires physical interaction, which makes them potentially vulnerable to side-channel attacks.

Another promising authentication method is the gaze-based approach. Improved eye-tracking techniques [17, 22] have enabled eye tracking on mobile devices such as smartphones and tablets. Gaze-based authentication methods do not require physical interaction [7–9, 12]. Moreover, they are touch-free, and thus are robust to various attacks such as shoulder-surfing attacks that peek at the

user from behind, smudge attacks that trace fingerprint smudges, and thermal attacks that detect heat after an operation. However, it has also been reported that a simple gaze trajectory or fixed position PIN/password information could be known if the attacker captures the eye movement [9]. In addition, *gaze passwords* have memorability concerns because they require the user to learn and remember unfamiliar password symbols [12].

Our research aims to enable robust authentication that is resistant to attacks from other people. To achieve this purpose, we propose an authentication method that uses a gaze pattern at meaningful keypoints in the image. Specifically, we utilize an ordered list of the names of keypoints as a secret (Fig. 1), and by gazing at each keypoint in order on a randomly displayed image, the user unlocks the device. Our method combines gaze-based authentication with graphical passwords, so we can expect it to exhibit both of their advantages. In addition, it provides robustness to observation attacks because it displays a different image for each authentication challenge, so the user gazes at different locations each time.

2 RELATED WORK

2.1 Graphical Passwords

Graphical passwords utilize visual memory and have many advantages in terms of memorability and usability over PINs/passwords with a sequence of characters or numbers [5]. They are also resistant to guessing attacks because the password space is theoretically larger. There are two types of graphical password authentication methods: selecting multiple images [5] and selecting multiple points in an image [3]. However, it has been reported that graphical passwords, which require physical input, are vulnerable to shoulder-surfing attacks [3].

2.2 Gaze-Based Authentication

There are two types of gaze-based authentication methods: one using *traditional passwords* such as entering a PIN/password by gaze input [6, 7, 14] and the other using *gaze passwords*. Gaze passwords utilize patterns of gazing at points displayed on the screen [19], predefined gaze trajectories (gaze gestures) [8], and free-form gaze trajectories [9]. Gaze-based authentication methods have the significant advantages of being robust against shoulder-surfing, smudge, and thermal attacks because they do not require physical interaction. However, they also have several problems. Entering traditional passwords by gazing has been reported to be slower in execution than finger-based input [12]. In addition, gaze passwords have the potential for memorability concerns because they require the user to learn and remember unfamiliar password symbols [12].

EyePassword [14] and EyePIN [7] achieve authentication by gaze typing instead of keyboard or touchpad typing. It has been reported that users prefer gaze typing to keyboard typing, especially in public places [14]. GazeTouchPIN [13] is a two-step authentication using touch-based PIN input and simultaneous gazing to the left and right, which is more secure against repeat and side-channel attacks. Free-Form Gaze Passwords [9] utilize user-defined gaze trajectories as gaze passwords that are not restricted to gaze points or gestures and have been shown to safeguard against shoulder surfing. However, it has also been reported that generating and executing gaze passwords in a completely free form can be uncomfortable for users [9]. Heikkila et al. introduced visual feedback to speed up gaze input and reduce the authentication time, but it may cause vulnerability to shoulder surfing attacks [18].

3 PROPOSED METHOD

We propose a gaze-based authentication method using graphical passwords consisting of keypoints in images without any additional hardware. We utilize graphical passwords as the secret and gaze as the input method. The authentication is based on a pattern of gazing at keypoints in the image. For example, in the case shown in Fig. 1(B), the four keypoints—left ear, left foot, right eye, and tail—are registered and stored in the order of gazing. The user unlocks the device by gazing at keypoints in the order of the registered secret on a randomly displayed image that is changed at regular intervals.

A keypoint is a characteristic point in an image, such as the annotated points in Fig. 1(B). Although such annotations usually refer to certain areas on the body, we define a keypoint here as a dot in an image represented by x and y coordinates. Different images of the same type (e.g., the cat and dog in Fig. 1(B)) can share the same keypoints. The user registers the names of keypoints and their order as a secret. Since multiple images can share keypoints, the secret, once registered, can be directly applied to another image. The actual keypoint locations are different for different images. By displaying a different image, the system can change locations to be gazed at without forcing the user to change his/her secret. Even if an attacker discovers the gazed locations based on the user's eye movements, the system remains secure so long as a cracked image is not used.

The user can remember the secret by looking at the image as well as the character string, which should enable the use of visual memory and may provide better memorability compared to just numerical or character string secrets such as PINs and passwords. In addition, gazing at a keypoint in the image allows the user to perform authentication by means of the natural behavior of looking at a specific point on the screen. Therefore, unlike gaze gestures, our method does not force the user to perform atypical actions.

The theoretical password space (TPS[1]) of our method depends on the number of keypoints available in the image (N_k) and the number of keypoints used as a secret (n), which is $\log_2 {N_k}^n$. For example, for the image shown in Fig. 1(B), the TPS is $\log_2 11^4 \approx 13.8$. Since our method does not have an obvious entry point like a dial pad or keyboard, an obvious location used for a secret is unavailable information for attackers. Therefore, the number of entry points available in an image depends on the eye-tracking resolution. Our preliminary experiment showed that the RMSE of eye tracking for the x-axis and y-axis was (44.82 pt, 62.28 pt), where 1 pt is approximately 0.17 mm. The screen size of 812 pt × 385 pt (iPhone 11 Pro) can contain 21 non-overlapping rectangles with the width = 44.82 pt and height = 62.28 pt. In this case, the TPS is $\log_2 21^4 \approx 17.6$, which can maintain sufficient robustness even with a small number of keypoints (e.g., $113.3 \approx \log_2 10^4$ for a 4-digit PIN and $20.7 \approx \log_2 64^4$ for an alphanumeric password). A shorter secret is preferable because it reduces the time for authentication.

[1]TPS grows exponentially and are typically compared in $\log_2$.

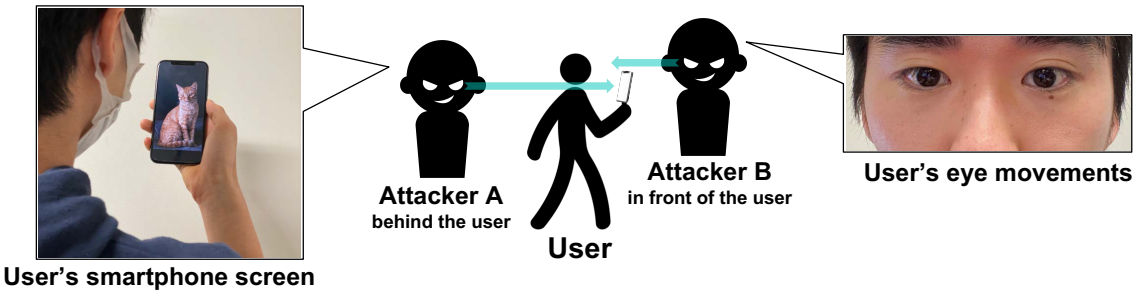

Figure 2: Information that an attacker can obtain. Attacker A behind the user can shoulder surf the smartphone screen (left). Attacker B in front of the user can observe the user's eye movements (right).

4 ROBUSTNESS AGAINST ATTACKS

Since our method combines graphical passwords with gaze-based authentication, it is expected to improve the robustness against attackers. Specifically, thanks to the characteristics of graphical passwords, our method is resistant to brute force attacks, and thanks to the characteristics of gaze-based authentication, our method is resistant to shoulder-surfing, smudge, and thermal attacks because there is no physical interaction.

We assume two attackers against our method, as shown in Fig. 2: A) those who try to shoulder surf against the smartphone screen, and B) those who try to read eye movements. Attacker A can look at the screen of the smartphone, but unlike with a PIN, the only accessible information is the image that the user is gazing at. Therefore, it will be difficult for the attacker to guess the secret. For Attacker B, although the attacker can observe the user's eyes, it has been reported that it is not easy to guess and accurately reproduce the user's gaze [9]. Even if the attacker finds out the approximate gaze movement, the gaze points can be changed by changing the authentication image to a different one while using the same secret. Thus, the user can maintain security without the burden of changing and remembering the secret.

Since the login interface of our authentication system just displays an image, the authentication procedure and the timing of the authentication itself are hidden. Therefore, it is not easy to recognize the user's actions even when observing the user. We expect that people who do not know this authentication will find it difficult to even try. In addition, it is not possible to make repeated tries on the same image because the image changes at regular intervals.

5 IMPLEMENTATION

In this study, for proof of concept, we implemented a prototype system on a smartphone, where two software modules were installed: an eye-tracking module and an authentication module. The system authenticates the user according to the user's gazing points.

5.1 Authentication Interface

The user follows the steps below to perform the authentication.

Registration The user selects multiple keypoints from the blue-marked points in the image displayed on the screen by touch input and registers the order of gazing as a secret (Fig. 3(A)(B)).

Authentication The user gazes at the keypoints corresponding to the secret in the registered order on randomly displayed images (Fig. 3(C)(D)). Blue points are not displayed during the authentication.

The user can perform registration and authentication on the screen shown in Fig. 3. For example, in Fig. 3(A), a dog is displayed, and the blue-marked keypoints are the candidates for a secret. The user selects four keypoints—*"right ear, front left foot, nose, and tail,"*—as a secret (Fig. 3(B)). At this time, the names of the registered keypoints are displayed on the screen for checking.

During authentication, a different animal image is randomly displayed (Fig. 3(C)(D)), and the user unlocks the device with the different gaze patterns by gazing at the corresponding keypoints in accordance with the registered secret. Vibrations are presented at each gaze as tactile feedback to inform the user that the gaze has been detected. An image for authentication challenge is displayed for a regular interval (currently, 10 seconds) and is changed when the authentication fails or when the regular interval passes. In addition, to prevent repeated tries by the attacker, authentication is disabled for 30 seconds after five failed tries (the typical limit of retries on a smartphone).

5.2 Eye-Tracking Module

We used ARKit[2] for eye tracking on the smartphone (iPhone 11 Pro[3]). ARKit can acquire information such as face orientation and facial expressions at about 60 fps using a front camera. The system calculates the intersections of the left and right gaze vectors and the screen of the device. It then calculates the midpoint between them as the user's viewpoint position

Preliminary experiments showed that our eye-tracking system causes an overall shift in the estimated viewpoint position. Therefore, we calibrate it before each authentication. The four corner points in the screen are used as reference points. The user gazes at each of the four points for two seconds. The system runs a regression analysis (SVR) for each x/y-axis using the viewpoint positions estimated from the data acquired while gazing as training data. The positions calculated by the regressor are used as the user's viewpoint position for authentication.

[2]https://developer.apple.com/documentation/arkit
[3]https://www.apple.com/jp/iphone-11-pro

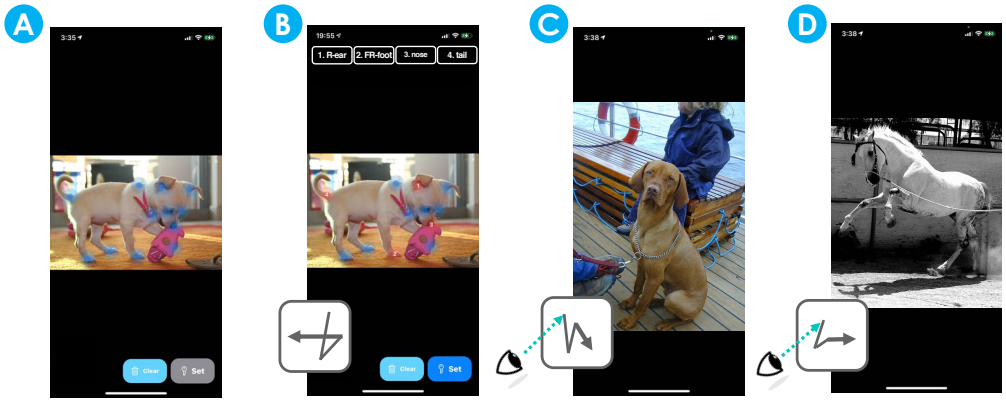

Figure 3: The screen for secret registration (A) before selecting and (B) after selecting. (C)(D) Samples of the authentication screen.

5.3 Authentication Module

Figure 4 shows an overview of the authentication algorithm. First, a randomly selected image is displayed on the screen (P1). The authentication system utilizes the user's eye-tracking position as input for authentication. In order to suppress the Midas touch problem [11], the system does not detect gaze input for 1 second after the image is displayed (P2), as the user searches for keypoints during this period. Once the system starts detecting gaze input, it checks if the sequence of gazed keypoints matches the secret or not, as explained in the next paragraph. If the authentication challenge fails, another image is displayed. When a regular interval (10 seconds) passes after the image is displayed, the image is changed (C1). When the number of failures reaches five, the system will display an image that cannot be used for authentication (i.e., not a similar image) for 30 seconds.

For the system to perform authentication, it is necessary to select keypoints based on the user's fixation and match the selected keypoint sequence with the registered secret. To obtain the user's fixation points, we used the Dispersion-Threshold Identification (I-DT) algorithm [20] with a time window (T) of 700 ms and a standard deviation threshold (th_{gaze}) of 10 pt. The system records the user's fixation point (G_i) (P3) and then, the system records the gazed keypoint (K_i) of the image within a fixed region ($G_x - d_x \leq K_x \leq G_x + d_x \wedge G_y - d_y \leq K_y \leq G_y + d_y$) from the output fixation point (C2). If multiple keypoints are detected, the point nearest to the fixation point is recorded (P4). Once a gazed keypoint is recorded, the recorded sequence of keypoint ($[K_1, \dots]$) is compared with the registered secret, and if the keypoints and their order of the two matches perfectly, the authentication succeeds (C3). If it matches the first part of the registered secret, the authentication continues with the same image (C4). If not, the authentication continues with the different image.

5.4 Image Data & Keypoints

In order to make this authentication system work, we need an image set with common keypoints and the location information of the keypoints in the images. We adopted the Animal-Pose Dataset for this requirement [2], one of the public datasets commonly used in the image recognition field. This dataset contains over 6000 images of cats, dogs, horses, sheep, and cows, including the locations of keypoints for each of the following nine categories: *Two eyes, Throat, Nose, Withers, Two Earbases, Tailbase, Four Elbows, Four Knees, and Four Paws*. We used seven of these, omitting *Four Elbows and Four Knees* because their distances to other points were too close.

6 EVALUATION

We conducted a preliminary evaluation of our prototype implementation. Five volunteers (P1–P5; 21–22 years old, mean = 21.4, 3 males) participated in the experiment, which was conducted indoors under fluorescent light. The participants sat on a chair and held an iPhone 11 Pro.

6.1 Tasks & Procedures

There are two tasks for the participants: secret registration and authentication execution. In the secret registration task, the participants register secrets to be used for authentication. Specifically, they select four keypoints from those displayed in the image by touch input and then decide on the order of the selected keypoints. In the authentication execution task, the participants gaze at the keypoints of the registered secret. The above two tasks are repeated five times as a set, and the participants perform four sets. Calibration is performed at the beginning of each set. After finishing, they were given a questionnaire consisting of three questions and 5-point Likert-scale answers (Tab. 1).

6.2 Results and Analysis

The experiment yielded the following data: $5_{\text{execution}} \times 4_{\text{sets}} \times 5_{\text{persons}} = 100_{\text{executions}}$. We also collected the results of the questionnaire (Tab. 1).

The acceptance rate was 86.0%. The number of authentication challenges per task for each participant is shown in Fig. 5. Within one retry, 95% of the authentication challenges were successful. These findings indicate that the acceptance rate needs to be improved. In particular, the confusion between left and right is considered to impact the acceptance rate strongly. We estimate that this is due to mis-recognition caused by the proximity of keypoints

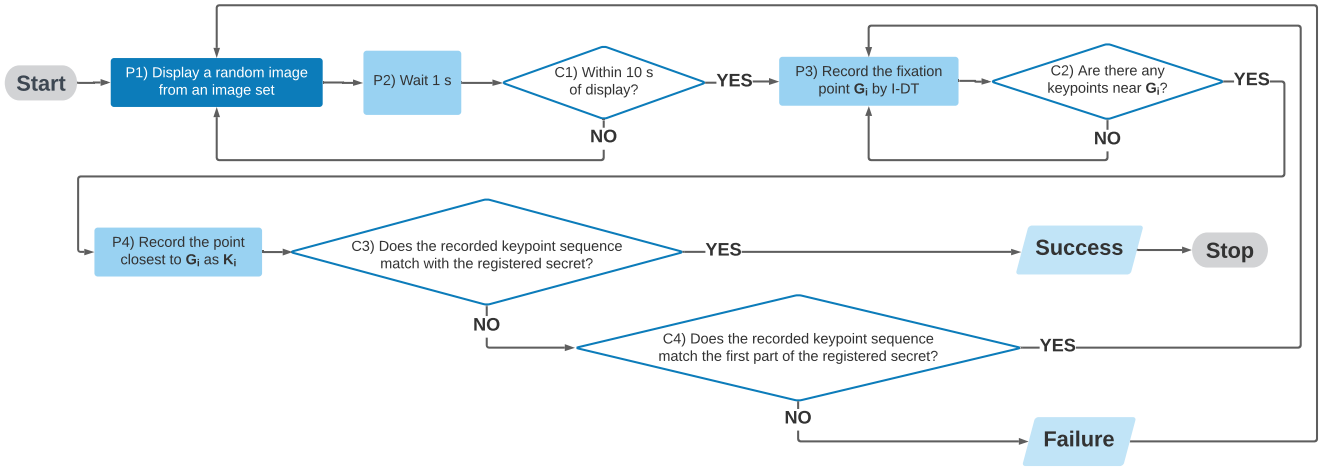

Figure 4: Flowchart of the authentication algorithm.

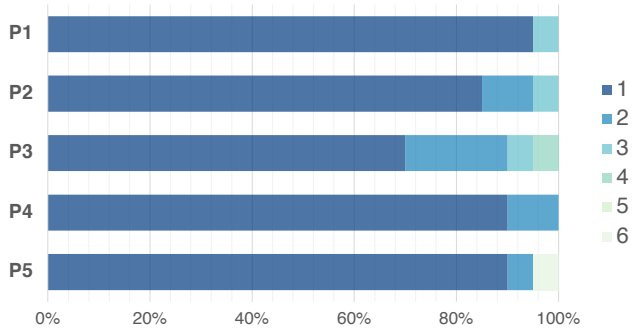

Figure 5: Number of authentication challenges per task.

with the left and right elements in the image, and mis-input caused by the user confusing the left and right keypoints and gazing at the wrong point.

To deal with mis-recognition, it is necessary to improve the accuracy of the eye tracking. We plan to introduce an eye-tracking technique based on deep learning, as proposed in [17, 22]. In addition, we plan to disable the selection of keypoints that are too close based on the RMSE of eye tracking. To deal with mis-input, we can use different image datasets such as landscape images or group photos of people, where left/right features for keypoints are scarce and multiple target objects are available.

The average execution time of the authentication was 5.03 s (SD=0.186 s). The execution time of the authentication is distributed within a very short range (Fig. 6), which means there is no large difference in the execution times for different executions or different users. Considering the results of our informal experiment with fewer keypoints authentication, the execution time of authentication approximately follows the product of a certain time required for the gaze detection and the number of keypoints. Therefore, it is possible to control the time required for authentication by decreasing the time for gaze detection and changing the number of keypoints in secret. As a final note, we should point out that the current system requires additional time for calibration. We are planning to use a calibration-free eye-tracking module in the future implementation.

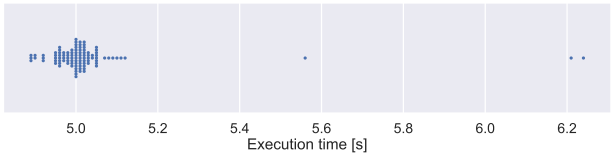

Figure 6: Distribution of authentication execution time.

The questionnaire items and answers are shown in Tab. 1, and the results for each user are shown in Fig. 7. Although the overall impressions of our method were favorable, some participants reported that they felt fatigued when executing authentication. This may have occurred because they were not familiar with gaze-based operations. On the other hand, there were several positive comments about the experience, such as *"I am glad that the authentication was smoother than I expected"* and *"It seems safe because it just looks like a screen saver is displayed."* The participants also reported that they selected 'obvious' keypoints such as nose and tail and no close keypoints. We need to investigate through further surveys whether there is any bias in the selected keypoints and whether there are any keypoints that attackers can easily break through.

7 CONCLUSION & FUTURE WORK

We proposed a gaze-based authentication method using graphical passwords consisting of keypoints in images. Thanks to combining gaze-based authentication with graphical passwords, our method is robust against shoulder-surfing, smudge, thermal, and brute force attacks. When unlocking the device, the user gazes at the keypoints in a randomly displayed image in accordance with the registered order. For proof-of-concept, we implemented an authentication

Table 1: Questionnaire presented to participants, with mean and std over Likert-scale answers.

No.	Questions + Answers	mean	std
Q1	How difficult did you feel it was to execute the authentication? Answers from (1) very difficult to (5) very easy.	4.2	0.84
Q2	Did you execute the authentication without feeling fatigued? Answers from (1) strongly disagree to (5) strongly agree.	3.6	1.14
Q3	Do you want to use our method? Answers from (1) strongly disagree to (5) strongly agree.	4.4	0.89

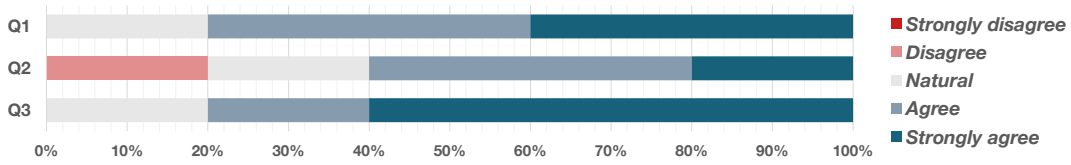

Figure 7: Questionnaire results.

system on a commercial smartphone where two software modules were added: an eye-tracking module and an authentication module that performs authentication according to the user's gaze point. No additional hardware was used. The results demonstrated the feasibility of our method.

In future work, we plan to improve the eye-tracking accuracy and conduct an attacker experiment to investigate the robustness against observation-spoofing attacks. We also plan to reexamine the authentication algorithm to verify the overall security of this authentication method. Additionally, we will apply our method to users' images (e.g., family photos) for detecting meaningful points using image recognition methods [4, 15, 21].

REFERENCES

[1] Mozhgan Azimpourkivi, Umut Topkara, and Bogdan Carbunar. 2017. A Secure Mobile Authentication Alternative to Biometrics. In *Proceedings of the 33rd Annual Computer Security Applications Conference* (Orlando, FL, USA) *(ACSAC 2017)*. Association for Computing Machinery, New York, NY, USA, 28–41. https://doi.org/10.1145/3134600.3134619

[2] Jinkun Cao, Hongyang Tang, Hao-Shu Fang, Xiaoyong Shen, Yu-Wing Tai, and Cewu Lu. 2019. Cross-Domain Adaptation for Animal Pose Estimation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (Seoul, Korea (South)). Institute of Electrical and Electronics Engineers, 9497–9506. https://doi.org/10.1109/iccv.2019.00959

[3] Andrew Lim Chee Yeung, Bryan Lee Weng Wai, Cheng Hao Fung, Fiza Mughal, and Vahab Iranmanesh. 2015. Graphical password: Shoulder-surfing resistant using falsification. In *2015 9th Malaysian Software Engineering Conference (MySEC)*. Institute of Electrical and Electronics Engineers, 145–148. https://doi.org/10.1109/MySEC.2015.7475211

[4] Grigorios G Chrysos, Stylianos Moschoglou, Giorgos Bouritsas, Jiankang Deng, Yannis Panagakis, and Stefanos P Zafeiriou. 2021. Deep Polynomial Neural Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP (Feb. 2021), 1–1. https://doi.org/10.1109/TPAMI.2021.3058891

[5] Antonella De Angeli, Lynne Coventry, Graham Johnson, and Karen Renaud. 2005. Is a picture really worth a thousand words? Exploring the feasibility of graphical authentication systems. *International Journal of Human-Computer Studies* 63, 1-2 (July 2005), 128–152. https://doi.org/10.1016/j.ijhcs.2005.04.020

[6] Alexander De Luca, Martin Denzel, and Heinrich Hussmann. 2009. Look into my eyes! can you guess my password?. In *Proceedings of the 5th Symposium on Usable Privacy and Security* (Mountain View, California, USA) *(SOUPS '09, 7)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/1572532.1572542

[7] Alexander De Luca, Roman Weiss, and Heiko Drewes. 2007. Evaluation of eye-gaze interaction methods for security enhanced PIN-entry. In *Proceedings of the 19th Australasian conference on Computer-Human Interaction: Entertaining User Interfaces* (Adelaide, Australia) *(OZCHI '07)*. Association for Computing Machinery, New York, NY, USA, 199–202. https://doi.org/10.1145/1324892.1324932

[8] Rainhard Dieter Findling, Le Ngu Nguyen, and Stephan Sigg. 2019. Closed-Eye Gaze Gestures: Detection and Recognition of Closed-Eye Movements with Cameras in Smart Glasses. In *Advances in Computational Intelligence (IWANN 2019)*. Springer International Publishing, Cham, Switzerland, 322–334. https://doi.org/10.1007/978-3-030-20521-8_27

[9] Eira Friström, Elias Lius, Niki Ulmanen, Paavo Hietala, Pauliina Kärkkäinen, Tommi Mäkinen, Stephan Sigg, and Rainhard Dieter Findling. 2019. Free-Form Gaze Passwords from Cameras Embedded in Smart Glasses. In *Proceedings of the 17th International Conference on Advances in Mobile Computing & Multimedia* (Munich, Germany) *(MoMM2019)*. Association for Computing Machinery, New York, NY, USA, 136–144. https://doi.org/10.1145/3365921.3365928

[10] Daniel Hintze, Philipp Hintze, Rainhard D Findling, and René Mayrhofer. 2017. A Large-Scale, Long-Term Analysis of Mobile Device Usage Characteristics. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 2 (June 2017), 1–21. https://doi.org/10.1145/3090078

[11] Robert J K Jacob. 1991. The use of eye movements in human-computer interaction techniques: what you look at is what you get. *ACM Transactions on Information and System Security* 9, 2 (April 1991), 152–169. https://doi.org/10.1145/123078.128728

[12] Christina Katsini, Yasmeen Abdrabou, George E Raptis, Mohamed Khamis, and Florian Alt. 2020. The Role of Eye Gaze in Security and Privacy Applications: Survey and Future HCI Research Directions. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–21. https://doi.org/10.1145/3313831.3376840

[13] Mohamed Khamis, Mariam Hassib, Emanuel von Zezschwitz, Andreas Bulling, and Florian Alt. 2017. GazeTouchPIN: protecting sensitive data on mobile devices using secure multimodal authentication. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction* (Glasgow, UK) *(ICMI '17)*. Association for Computing Machinery, New York, NY, USA, 446–450. https://doi.org/10.1145/3136755.3136809

[14] Manu Kumar, Tal Garfinkel, Dan Boneh, and Terry Winograd. 2007. Reducing shoulder-surfing by using gaze-based password entry. In *Proceedings of the 3rd symposium on Usable privacy and security* (Pittsburgh, Pennsylvania, USA) *(SOUPS '07)*. Association for Computing Machinery, New York, NY, USA, 13–19. https://doi.org/10.1145/1280680.1280683

[15] Alexander Mathis, Pranav Mamidanna, Kevin M Cury, Taiga Abe, Venkatesh N Murthy, Mackenzie Weygandt Mathis, and Matthias Bethge. 2018. DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nature neuroscience* 21, 9 (Sept. 2018), 1281–1289. https://doi.org/10.1038/s41593-018-0209-y

[16] Wendy Moncur and Grégory Leplâtre. 2007. Pictures at the ATM: exploring the usability of multiple graphical passwords. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '07)*. Association for Computing Machinery, New York, NY, USA, 887–894. https://doi.org/10.1145/1240624.1240758

[17] Joonbeom Park, Seonghoon Park, and Hojung Cha. 2021. GAZEL: Runtime Gaze Tracking for Smartphones. In *2021 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. Institute of Electrical and Electronics Engineers, 1–10. https://doi.org/10.1109/PERCOM50583.2021.9439113

[18] Räihä, Kari-Jouko and Heikkilä, Henna. 2009. Speed and Accuracy of Gaze Gestures. *Journal of Eye Movement Researc* 3, 2 (November 2009). https://doi.org/10.16910/JEMR.3.2.1

[19] Vijay Rajanna, Seth Polsley, Paul Taele, and Tracy Hammond. 2017. A Gaze Gesture-Based User Authentication System to Counter Shoulder-Surfing Attacks. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI EA '17)*. Association for Computing Machinery, New York, NY, USA, 1978–1986. https://doi.org/10.1145/3027063.3053070

[20] Dario D Salvucci and Joseph H Goldberg. 2000. Identifying fixations and saccades in eye-tracking protocols. In *Proceedings of the 2000 symposium on Eye tracking research & applications* (Palm Beach Gardens, Florida, USA) *(ETRA '00)*. Association for Computing Machinery, New York, NY, USA, 71–78. https://doi.org/10.1145/355017.355028

[21] Mingxing Tan, Ruoming Pang, and Quoc V Le. 2020. EfficientDet: Scalable and Efficient Object Detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Seattle, WA, USA). Institute of Electrical and Electronics Engineers, 10778–10787. https://doi.org/10.1109/cvpr42600.2020.01079

[22] Nachiappan Valliappan, Na Dai, Ethan Steinberg, Junfeng He, Kantwon Rogers, Venky Ramachandran, Pingmei Xu, Mina Shojaeizadeh, Li Guo, Kai Kohlhoff, and Vidhya Navalpakkam. 2020. Accelerating eye movement research via accurate and affordable smartphone eye tracking. *Nature communications* 11, 1 (Sept. 2020), 4553. https://doi.org/10.1038/s41467-020-18360-5

[23] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. 2005. PassPoints: design and longitudinal evaluation of a graphical password system. *International Journal of Human-Computer Studies* 63, 1-2 (July 2005), 102–127. https://doi.org/10.1016/j.ijhcs.2005.04.010