Saarland University
Human Computer Interaction
Department of Computer Science

Bachelor Thesis

Development and Evaluation of Authentication Schemes for Mobile Virtual Reality

Daniel Schäfer

March 22nd, 2018

Advisor
Marco Speicher

Supervisor
Prof. Dr. Antonio Krüger

Reviewers
Prof. Dr. Antonio Krüger
Dr. Florian Daiber

Declarations

Eidesstattliche Erklärung / Statement in Lieu of an Oath:
Ich erkläre hiermit an Eides Statt, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.
I hereby confirm that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis.

Unterschrift / Signature:

Daniel Schäfer

Einverständniserklärung / Declaration of Consent:
Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der Informatik aufgenommen und damit veröffentlicht wird.
I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

Unterschrift / Signature:

Daniel Schäfer

Abstract

In recent years Virtual Reality has grown very quickly and found more and more use-cases. With new commercial releases of affordable mobile HMDs such as Google Daydream, Samsung Gear VR combined with the rising potential of mobile AR glasses, the unsolved security problem for these devices grows continuously. Existing knowledge-based and biometric authentication schemes all come with critical disadvantages especially because they are usually not designed with Virtual Reality usage in mind. I implemented a novel graphical authentication scheme called "Personal Environment Authentication" specifically aimed at Virtual Reality. In this new scheme the user will authenticate herself using an ordered selection of objects in a (personal) virtual environment in form of of a 360-degree image. It offers significantly improved memorability of passwords compared to knowledge-based systems making use of the Pictorial Superiority Effect. In a study I was able to additionally confirm significantly improved user-experience and security of this scheme compared to the prominent solutions "PIN" and "Android Pattern" authentication.

Contents

1 Introduction 8
1.1 Motivation . . . 8
1.2 Research Goals . . . 9
1.3 Significance of the Study . . . 10
1.4 Outline . . . 10

2 Related Work 11
2.1 Virtual and Augmented Reality . . . 11
2.2 The perfect Virtual Reality . . . 12
2.3 Existing Authentication Methods . . . 12
2.4 Why is improvement so important? . . . 16
2.5 Advantages of Graphical Authentication Schemes . . . 19
2.6 Adapting Established Approaches to Virtual Reality . . . 23

3 Concept 24
3.1 Foundation . . . 24
3.2 Overview . . . 25
3.3 Design . . . 26
3.4 Interactions . . . 28

4 Implementation 29
4.1 System Overview . . . 29
4.2 Implementation . . . 31
4.3 Architecture . . . 38
4.4 Additional Security Enhancements . . . 40

5 User Study 41
5.1 Goals of the Study . . . 41
5.2 Participants . . . 41
5.3 Experiment Setup . . . 42
5.4 Study Design . . . 43
5.5 Procedure . . . 45
5.6 Attacker Procedure . . . 47
5.7 Hypothesis . . . 47

6 Results 48
6.1 Registration and Login Time . . . 48
6.2 User Errors . . . 50
6.3 Attacking the Authentication Schemes . . . 51
6.4 User Experience . . . 54

6.5 Workload . . . 57
6.6 Demographics . . . 58

7 Discussion 60
7.1 Registration and Login Time . . . 60
7.2 User Errors . . . 61
7.3 Attacking the Authentication Schemes . . . 62
7.4 User Experience . . . 64
7.5 Workload . . . 64
7.6 Demographics . . . 65

8 Conclusion 66

9 Future Work 67

Acknowledgement 70

List of Figures 71

Abbreviations 73

Bibliography 74

Appendices 78

1 Introduction

1.1 Motivation

In recent years Virtual Reality (VR) has found more and more use cases. The most prominent ones being the entertainment industry with especially gaming and movie consumption, visualization of complex shapes and scientific research.[1] The advantages of a Virtual Reality are quite obvious. The interaction with the surroundings is more natural than it has ever been before on a computer system. The user can quite literally interact with virtual objects exactly the way she would in the real world which provides a strong immersion for any user while being intuitive and natural to use. [1]

Recently VR devices have become more accessible to consumers. The initial price for hardware to 'enter' a virtual world is very low compared to a couple of years ago. But not only the price has dropped significantly, the hardware and software associated with it has become easier and more comfortable to use. Also the big disadvantage of 'motion sickness' has become much less of a problem, thanks to considerably increased performance of HMDs and sensors. [2] The hardware from today, combined with much improved sensors, built into consumer hardware nowadays are not comparable to the hardware from ten years ago.

Augmented Reality (AR) also is on the rise, with projects like the Google Glass project which caught a lot of public attention in recent years. It seems realistic to expect that Augmented and Virtual Reality are going to develop further along its trend. They might eventually become an essential part of our daily lives, just like smartphones 'recently' have. According to forecasts the Augmented / Virtual Reality industry could become a close competitor to the mobile industry in the near future already.[2]

According to industry analyst International Data Corporation (IDC), the AR/VR headset market is predicted to continue its growth reaching 81.2 million units of VR headsets shipped until 2021. This would result in an annual growth rate of around 56%.[3] Many experts are expecting AR to match or even surpass VR revenue in the next ten years.[2]

[1] teslasuit.io/blog/virtual-reality/111-virtual-reality-applications-use-cases/ visited 16.03.18

[2] www.digi-capital.com/news/2017/01/after-mixed-year-mobile-ar-to-drive-108-billion-vrar-market-by-2021/ visited 21.02.18

[3] www.forbes.com/sites/paullamkin/2017/09/29/vr-and-ar-headsets-to-hit-80-million-by-2021/ visited 21.02.18

But with all this predicted growth the problem of security arises. It is not acceptable that anybody is able to take someones AR/VR headset or glasses and access sensible information without any kind of authentication necessary. The field of authentication in VR has stayed largely untouched and my thesis aims to address this problem.

The main issue I am trying to solve with this thesis is secure user authentication in the Virtual Reality. I will develop a new authentication scheme, aimed at Virtual Reality which will also be transferable to Augmented Reality. My scheme will try to optimize the user experience in Virtual Reality, while offering improved security and memorability compared to already established measures of authentication.

1.2 Research Goals

The goal of my thesis is to answer the following **Research Questions**:

- Does 'Personal Environment Authentication' provide a better user experience in VR than established authentication methods?
- Does 'Personal Environment Authentication' offer enhanced security against shoulder surfing attacks compared to established authentication methods.
- Do 'Personal Environment Authentication'-passwords offer increased memorability compared to alphanumeric passwords?

To answer these questions I conducted a study which is described in section 5 of my thesis. This study observed the behavior of participants in both the 'PIN' and 'Android Pattern' authentication scheme compared to the behavior in my newly developed 'Personal Environment Authentication' scheme. To achieve this I logged the time it took to perform login/registration and the occurrence of user-errors amongst other measures. Additionally I provided questionnaires which go into detail about the experience and impressions the participants had using the three different authentication schemes.

To observe the security of the schemes, the participants were filmed from the outside while performing authentications with randomly generated passwords in all three authentication schemes. These videos were watched by an experienced and extensively trained attacker who tried to authenticate himself using the information he gained from watching the participant login. I will provide detailed statistics of how likely it was for an attack to be successful in what amount of time depending on the length of the password and the used authentication scheme to compare the security of the three authentication methods.

1.3 Significance of the Study

My newly developed Personal Environment Authentication scheme implements a graphical authentication scheme optimized for usage in VR. The study proved that my approach offers an improved user experience compared to the established authentication schemes it was compared with.

The only disadvantage discovered in my study was the significantly higher workload and necessary time to register/login in the Personal Environment Authentication scheme. This can partially be explained by the randomness of passwords that were used in the experiments. Additionally all the participants were completely inexperienced in my novel scheme, often leading to slower movement and reactions.

However the study proved that both PIN and Android Pattern authentication methods are extremely insecure against an experienced attacker, while the Personal Environment Authentication scheme offered a much higher security even with much shorter password length.

1.4 Outline

The following related work chapter will introduce many ideas that contributed to the concept of my Environment Authentication scheme. I will also define relevant terms that will be used throughout my thesis. Additionally I will provide an overview over different existing methods of authentication with their most prominent advantages and disadvantages.

This will serve as a starting point to explain what the main issues of existing schemes in the context of VR are.

In the following concept chapter I will explain the designing process and ideas behind the environment authentication scheme, followed by a detailed description of problems solved during implementation in section 4.

Afterwards I will discuss the procedure and results of the conducted study followed by a conclusion of this thesis. In the last chapter I will provide an outlook to future work that could be done to extend the findings of my thesis.

2 Related Work

2.1 Virtual and Augmented Reality

Humans perceive the world over multiple different senses. The human retina captures the reflected light by objects which triggers various reactions in the human brain depending on the object. Virtual Reality (VR) mimics these stimuli in the human brain which occur in the real world very closely using a stereoscopic three-dimensional rendering inside a Head-Mounted-Display (HMD). [1]

The HMD displays separate pictures for both eyes which makes it possible to achieve a very natural perception of three-dimensional space and objects using just visual senses. The technique used to render these pictures for both eyes is called stereoscopic three-dimensional rendering. This visual perception paired with realistic (haptic) feedback makes for a very immersive experience. Most HMDs e.g. offer a gyroscope / accelerometer used to transfer the users head movement one-to-one into the virtual world. In most scenarios the user additionally uses a controller which transfers movement of her hand(s) into the virtual world as well. [1]

While VR captures the user's entire range of vision, Augmented Reality (AR) only adds additional information to the natural field of view. A very well known example for this is the Google Glass project, giving their user notifications when they receive e.g. an email. This notification will be displayed immediately in the peripheral vision of the user - so the user saves herself some time, not having to take her phone out to look at the phone's display. [1]

Both VR and AR offer ways of human-computer interaction which have not been possible before. They can make complex information easier to understand and more approachable because one can e.g. directly visualize a building's blueprint in three-dimensional space and interact with it. This level of interaction would obviously be impossible for a two-dimensional map. [3] VR and AR also enable advanced simulators for race drivers or pilots saving not only money, wear and pollution but also possibly human lives. [4] But there are also much less extreme use-cases in daily life, especially for AR. E.g. most features a smartphone offers today could be solved in AR using more natural and intuitive interactions while also being less physically demanding. That means the user does not have to take her phone out of her pocket to see who wrote an email or to simply check the time or date which would occupy at least one hand. In certain situations, like driving a car, this advantage of 'having free hands' should be valued very highly.

2.2 The perfect Virtual Reality

Ralf Dörner defines the perfect Virtual Reality with an artificially-generated world which is able to stimulate all human senses in the same quality and quantity as a human is used to in the real world. Additionally it is just as important that the computer-generated world reacts precisely like the real world would. That means that a user can e.g. grab a glass of water in the computer-generated world using her hand and would also feel the glass with the cold water inside itAll actions and sensations possible in the real world would identically transfer to the perfect virtual world. [5]

Obviously the VR systems from today are far from perfect, but the target clearly is to improve upon all these qualities with improvements in both software and hardware.

2.3 Existing Authentication Methods

There are many different approaches to solve the problem of authentication. Most of them can be applied to a wide range of devices, including VR devices. In the following paragraph I am going to divide existing authentication methods into five main categories:

Knowledge-based Authentication Methods are by far the most commonly used authentication schemes today across a wide range of devices. It describes all authentication schemes which involve the exact recall of some particular information. The most prominently used implementation of this are Personal Identification Numbers, also called PINs and alphanumeric passwords. The biggest disadvantage of all these approaches is the compromise between security and memorability of mentioned passwords or PINs. To be considered secure these codes should be both long and seemingly random which at the same time would also make them very hard to remember. [6] Simply writing the passwords down might solve the problem on first glance, but one would have to store this information somewhere secure which is another problem in and of itself.

The issue of remembering passwords has gotten a lot worse in recent years, because it has become increasingly important to use unique passwords for every service one is authenticating on. This is mainly caused by the substantial number of large-scale security breaches in which passwords of millions of users were stolen from large databases online. The worst known incident was the Yahoo breach in 2013 when the passwords (and other sensible data) of around three billion users were compromised at once. While the passwords were encrypted, Yahoo was using an outdated encryption which was at the time already easy to crack.[4] Assuming one uses the same password on multiple services,

[4] `reuters.com/article/us-yahoo-cyber/yahoo-says-all-three-billion-accounts-hacked-in-2013-data-theft-idUSKCN1C8201/` visited 14.02.18

a security breach of just one of these services might cause loosing access to all of them, giving the attacker access to all the personal information in these services.

In general alphanumeric passwords have proven to be predictable whenever the user is allowed to choose the password herself. [7] This is the main reason why dictionary attacks are the most prominently used attack against an alphanumeric user-created password. A dictionary attack is a modification of a simple brute-force attack which e.g. brute forces the ten million most used alphanumeric passwords exploiting the predictability of user chosen alphanumeric passwords. Nowadays there are much more efficient and intelligent dictionary attacks out there which go far beyond brute-forcing the ten million most used passwords. [8]

Biometric Authentication Methods make use of unique physiological characteristics of a human. These schemes are already fairly established, most prominently in smartphones which make use of fingerprint scanners, iris scanners or authentication using facial features. [9]

Although all these methods are generally easy to use and avoid the problem of information recall entirely, all biometric authentication schemes share a very large drawback. Once biometric information has been compromised one can not simply change it like an alphanumeric password. The collection of these biometric informations by government agencies and other companies expose a huge risk to this unchangeable data. [9]

Gesture Authentication Methods e.g. the Android Pattern Authentication scheme is an adaption to an alphanumeric PIN code which is mainly used on Android smartphones nowadays. The scheme is very adaptable in the sense that the user can create simple but also fairly complex gestures. Introducing the human motor memory can make this gesture secret easier to recall compared to an ordinary alphanumeric PIN. Gesture authentication is also very flexible in terms of user-input source. Everything ranging from a simple touchscreen to a acceleration sensor mounted to a hand can be used to perform a gesture. [10]

There are already some working authentication schemes for VR developed making use of gestures, e.g. KinWrite by Tian et al. [11] which authenticates the user by observing a handwriting motion using Microsoft Kinect as shown in figure 1.

The scheme was able to achieve 100% precision and 70% recall in the worst case. Unfortunately an approach like this is not feasible for 'mobile' VR as specialized sensors to capture the user's hand movement (in this case Microsoft Kinect) are required.

A much more mobile approach of the same idea was implemented and tested by Roshandel et al. by using a finger ring which contains multiple sensors as shown in figure 2 to capture the hand and finger movement. [12] Just like with KinWrite, the user authenticates by drawing a signature in the air. The experiments in this paper showed that

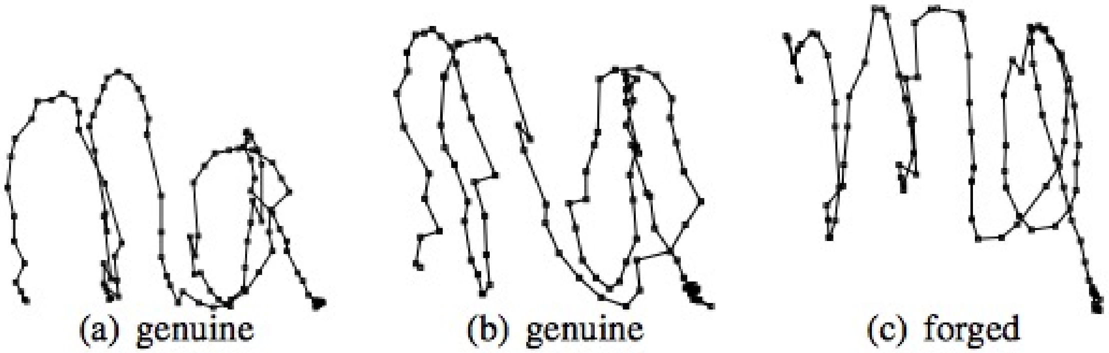


Figure 1: *KinWrite successfully verified the genuine signature and rejected the forged handwriting. [11]*

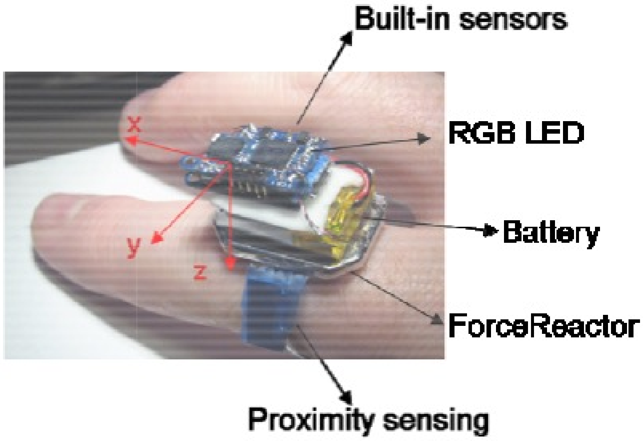


Figure 2: *Prototype of the multi-sensor based framework called Pingu. [12]*

signature recognition using simple classification algorithms can reach a very high accuracy. [12]

While this approach is clearly much more mobile than Microsoft Kinect, it still requires additional hardware for authentication which is a significant disadvantage to a universally applicable approach without the need of any additional hardware.

Graphical Authentication Schemes are based on the assumption that pictures are generally easier to remember than words or numbers while also offering improved security. The improved memorability is caused by the so called Pictorial Superiority Effect [13] and the increased security because of the high complexity a picture offers compared to alphanumeric codes among others. [14]

There are many different theories which explain this phenomena, the most widely accepted theory is the so called 'dual-coding theory' by Paivio. [15] It suggests that the brain processes and represents all verbal and non-verbal memory differently. Image-based memories retain their perceptual features in their mental representation and are assigned to a perceived meaning rather than pure observation. [16] On the other hand text-based information is represented symbolically in which the "symbols are given a meaning cognitively associated with the text, as opposed to a perceived meaning based on the form of text.[..] This additional processing required for verbal memory renders this a more difficult cognitive task."(2012, Biddle et al.) [17]

Graphical authentication schemes can be based on recognition or recall. [14] An example for a recognition-based graphical authentication scheme which inspired my approach is the "Story" scheme developed by Davis et al. [18]. In this scheme the user chooses the password pictures from nine distinct 'every-day' categories being: animals, cars, women, food, children, men, objects, nature, and sports. The user must authenticate by selecting the same images already selected in the registration among dummy images.

The recall-based graphical authentication scheme which inspired the approach I took to develop my novel authentication scheme is the "PassPoints" scheme developed by Wiedenbeck et al. in which the user authenticates by clicking on previously selected areas of a user-selected picture. [19, 20]

Both the PassPoints and the Story scheme will be explained in more detail in a following paragraph.

Hardware-based Authentication also called object-based is an authentication that relies on physical access to some different hardware. A prominently used example of this is two-factor authentication which relies on a one-time password randomly generated on a device only the user has access to (e.g. the user's smart-phone). This password is then used to authenticate on some service in addition to or instead of another authentication method. [21] Another rarely used approach is a physical cryptographic key e.g. stored on a USB stick which has to be inserted to authenticate. [22]

All these different approaches come with a variety of advantages and disadvantages. None of the categories are the 'perfect choice' for all scenarios and use cases and generally there are a lot of good reasons to develop new authentication schemes to maybe replace commonly used approaches from today.

2.4 Why is improvement so important?

The following section of related work contains general problems of the currently most used authentication schemes and explains why it is absolutely necessary to improve the security of all devices accessible to strangers by inventing alternative authentication schemes.

Is Secure and Usable Smartphone Authentication asking too much?

The publication “is secure and usable smartphone authentication asking too much” summarizes these core problems of current schemes and the importance of new secure authentication schemes for everyone’s devices on the example of smartphones. [23] With increasing functionality of smartphones in the recent years, the most impactful ones being the introduction of mobile banking, social networks and various communication channels, there is also a growing need for security. Nowadays an attacker is not only able to acquire all sorts of personal and financial data stored on the device but also everything which is accessible through the internet using the device and all applications installed. Although the importance of a secure device has increased tremendously in the past years most people are still using very outdated authentication schemes with one of the most used ones being simply a four-digit number. Alexander De Luca and Janne Lindqvist summarized alternative solutions to improve various authentication problems from the fields of biometric, gesture-based and hardware-based authentication. [23]

Although this publication talks about authentication schemes mainly aimed at smartphones, it is still very relevant for my thesis because the core problems, a secure authentication scheme is facing, are transferable to VR. Over the next years VR and AR technology is expected to grow at a very rapid pace[5] and with improved technology, there might be scenarios in which VR/AR glasses replace a smartphone for specific use cases. Just as with smartphones today, a secure and user-friendly authentication scheme is going to be even more important given the growing possibilities of AR and VR devices.

An Efficient Mechanism for Secure Authentication

This paper published in 2013 by Sharikala et al. gave me the initial idea of how I would like to approach and improve the currently used authentication systems. The general idea of the paper combines features of the Deja Vu, Cued Click Points, the Secret Draw technique and text passwords to improve the security of user authentication on mobile devices. A user who demands authentication has to select specific points in a series of pictures and add text to each selected point. After that she has to draw a secret

[5] forbes.com/sites/paullamkin/2017/09/29/vr-and-ar-headsets-to-hit-80-million-by-2021 visited 21.02.18

on another picture and only if both these steps were performed correctly the user will successfully authenticate to the system. [24]

Sharikala M. Deshmukh and P.R. Devale developed this scheme by looking at all the currently used approaches and their strengths and weaknesses regarding security. The authentication system explained above is the solution they proposed to solve the main problems being: incremental guessing attacks, hotspots, false accept, predictable passwords, and easy guessing in general. [24]

The concept of combining different authentication approaches to decrease the risk of an attacker being able to authenticate as somebody else is a very promising approach.

GazeTouchPass: Multimodal Authentication Using Gaze and Touch on Mobile Devices

A different approach to increase the security of authentication scheme is presented in the publication “GazeTouchPass, a multimodal authentication scheme“ developed by Khamis et al. The publication focuses on combining different input-modalities during the authentication process to improve security, especially against shoulder surfing attacks. The scheme accepts input via touch on a 10-digit keypad and gazing to the left or right as part of a password sequence, e.g. `3-left-4-right` could represent a user’s password as shown in figure 3. [25]

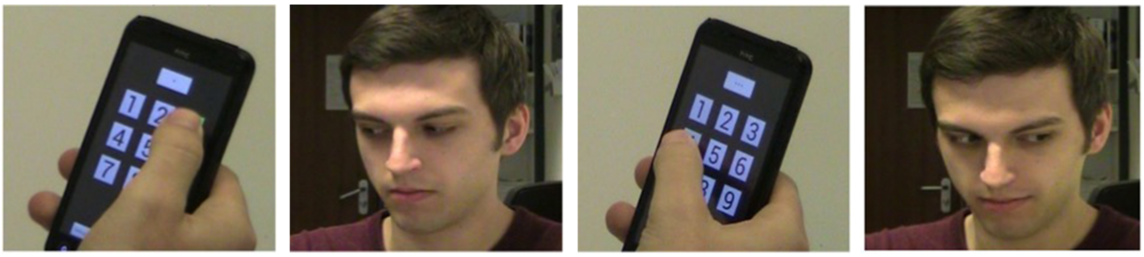
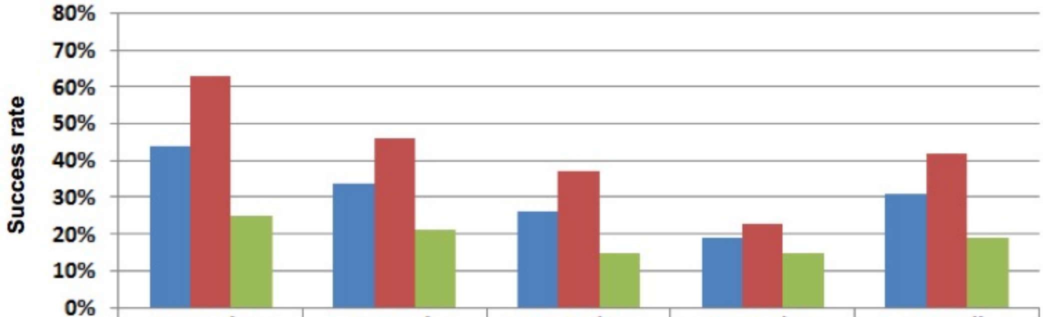


Figure 3: *Password sequence 3-left-4-right. [25]*

In their study they investigated how easy it is for an shoulder surfing attacker who knows the authentication scheme to guess the password after observing both head-movement and the screen during a successful login process of a user.



	0-switches	1-switch	2-switches	3-switches	Overall
🟦 All Attacks	44%	34%	26%	19%	31%
🟥 Iterative Attacks	63%	46%	37%	23%	42%
🟩 Side Attacks	25%	21%	15%	15%	19%

Figure 4: *Success rate on attacking GazeTouchPass which shows the decreasing success rate with increasing the number of modality switches. [25]*

The experiment showed very clearly that the success rate of an attack decreases significantly the more often the input modality was switched for a specific password as can be seen in figure 4. That means that e.g. `3-left-4-right` (3 input modality switches) was much harder to guess for an attacker than `3-4-left-right` (only 1 input modality switch) even with the attacker being able to observe the login multiple times before performing his attack. [25]

While the experiments showed that this scheme is clearly not the perfect solution, it displayed the security improvement multimodal authentication methods can achieve while maintaining high usability which is one of the most important aspects of authentication methods. The reason why most people are using insecure four digit codes as authentication is simply because all alternatives are either too much of a usability downgrade for them or because they have problems remembering longer passwords. The security improvement multimodal authentication schemes provide is very valuable information that can be translated to all kinds of different authentication methods.

2.5 Advantages of Graphical Authentication Schemes

There are a lot of good reasons to use graphical passwords as opposed to knowledge-based authentication schemes or even biometric approaches. In general graphical authentication schemes are able to solve multiple problems the currently most prominently used schemes are facing. In the following section I will outline advantages and possibilities of these schemes in the field of VR and AR.

Is a picture really worth a thousand words? Exploring the feasibility of graphical authentication systems

The publication "Is a picture really worth a thousand words? Exploring the feasibility of graphical authentication systems" by Coventry el al. summarizes the reasons why I decided to develop a graphical authentication scheme for AR/VR devices instead of a text-based authentication scheme which are significantly more popular. [26]

Knowledge text-based authentication schemes require a tradeoff between security and the human memory constraints. [27] On the other hand graphical authentication schemes substitute the need to remember a precise alphanumeric code with the recognition of previously learned pictures which is a task the human brain is much more proficient at, shown in various cognitive psychology studies (Nickerson 1965, Shepard 1967, Standing et al. 1970, Standing 1973). This means that it is not only easier for a human to remember the pictures but they are also remembered for a longer period of time, which is called the Pictorial Superiority Effect. [13, 28]

Designing a good graphical authentication mechanism is a complex task, because it requires the weighting of various different factors to maximize usability while maintaining high security. This paper describes in detail why and how certain factors may influence the usability of a graphical authentication scheme.

Another obvious advantage of graphical authentication schemes in the context of VR is that VR and AR devices primary purpose is to enhance or create a world surrounding the user. This property makes pictures in VR not only appear more natural compared to text, but also adds the possibility of three-dimensional pictures which suits the VR approach very well. These three-dimensional pictures could improve the security of graphical authentication schemes even further because the third dimension is adding a lot of information to a picture which can effortlessly be processed by the human brain because we are projecting two dimensional pictures into the three-dimensional space anyways. [1]

Improving user authentication on mobile devices: A Touchscreen Graphical Password

Especially in smartphones and tablets one of the widely used input modalities is touchscreen input which was used in the publication “Improving user authentication on mobile devices: A Touchscreen Graphical Password” by Chiang et al. to design a new graphical password scheme which they called “Touchscreen Multi-Layered Drawing” or short TMD.

This scheme was able to significantly improve memorability and decrease error rate compared to a commonly used graphical authentication scheme called “Draw A Secret”. I am going to use the observations and the user feedback which was provided by 90 participants on this particular scheme to improve the first concept of my novel authentication scheme. [29]

I think that many design improvements they listed in this paper are not limited to authentication methods using touchscreens, but can be equally transfered to schemes using other types of user-input, e.g. head and controller movement in VR. [29]

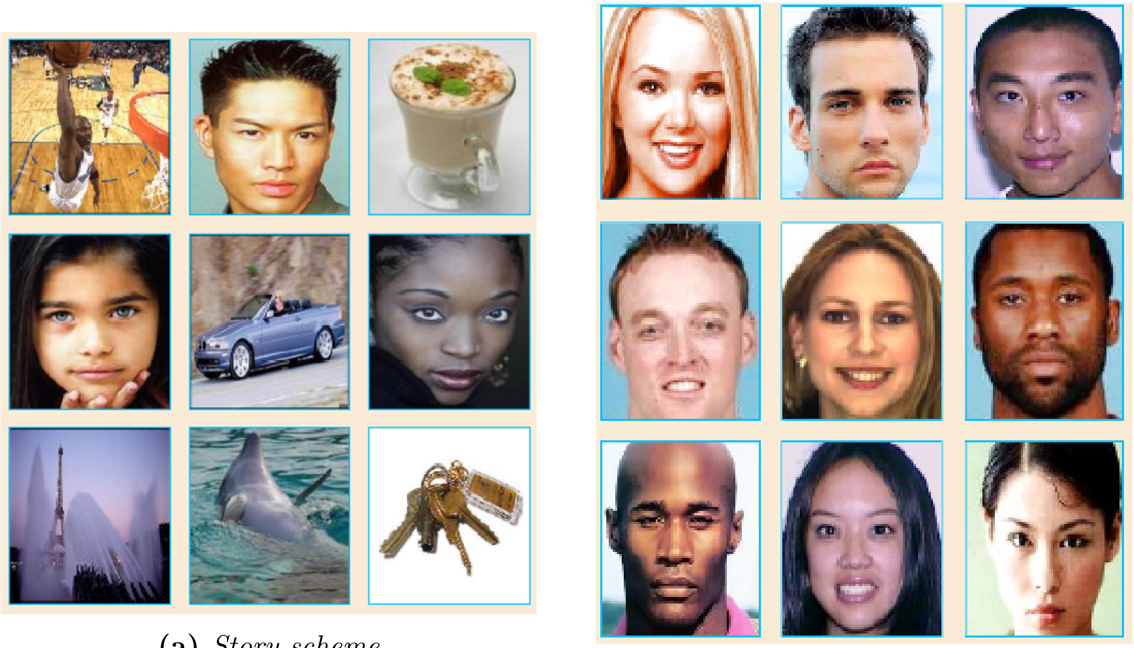
Universal Multi-Factor Authentication using graphical Passwords

Another very interesting approach when using graphical passwords is combining them with multi-factor authentication just like Sabzevar et. al did in their publication “Universal Multi-Factor Authentication using Graphical Passwords”. For this scheme the user is challenged by the service provider with an image where she has to click certain points in a certain order. To solve this task the user receives a message containing necessary hint information on his hand-held device. [30]

With two-factor authentication being a widely adopted security feature nowadays an idea like this could obviously improve the security of graphical authentication schemes used in the AR or VR even further. It could modify known pictures or the order the user has to click these certain points and give only the device owner access to the necessary hint to solve the modified scheme. Overall this is a very interesting approach to optionally add to a scheme to make it even more secure, if the user wants to, at a fairly small expense of usability.

On User Choice in Graphical Password Schemes

Davis et al. developed a novel graphical authentication scheme called ’Story’ which takes advantage of the biggest strength of graphical passwords chosen by users. It authenticates the user with a sequence of images selected by the user forming a ’story’. The selected pictures are each from a set of nine every-day categories being: animals, cars, women, food, children, men, objects, nature and sports as seen in figure 5(a). [18] The idea of this authentication scheme is to increase memorability of passwords using a story which "connects" the different pictures.



(a) *Story scheme*

(b) *Face scheme*

Figure 5: *Example selection of both authentication schemes. [18]*

Davis et al. were not able to prove a higher memorability rate when compared to a scheme called ’Face’ (which was developed similar to the commercial ’Passfaces’ scheme) in their study [18]. The Face scheme worked exactly like the Story scheme, but uses exclusively pictures of faces to authenticate as shown in figure 5(b). The used pictures for both schemes were pre-selected, so the user had to think of and remember a new story and was not able to use a story he experienced himself in the past with original pictures of the objects/persons contained in this story which would have most likely increased memorability significantly. Generally the memorability of passwords was still good in this authentication scheme, and they were able to significantly increase security when compared to the Face authentication scheme. [18]

PassPoints: Design and longitudinal evaluation of a graphical password system

An authentication scheme which is not limited by the usage of pre-selected images is the PassPoints scheme developed by Wiedenbeck et al. [19] This graphical authentication scheme allows the user to authenticate using a picture which was selected by herself in the registration phase. The user has to repeat an ordered sequence of clicks on coordinates in the image correctly to login as shown in figure 6. The scheme offers very high flexibility, as there are no predefined areas in the picture and no artificial boundaries.

Assuming a reasonable choice of the picture, which should contain a sufficient amount of details, the password space on the picture is very large compared to a numeric or even alphanumeric password. [19] In a login process the user has to repeat the sequence of clicks on coordinates in the image he entered during registration. Every click during login has a tolerance of a small, fixed number of pixels which means that the user does not have to click on the exact same pixel, but only a pixel reasonably close to it, as indicated in figure 6.

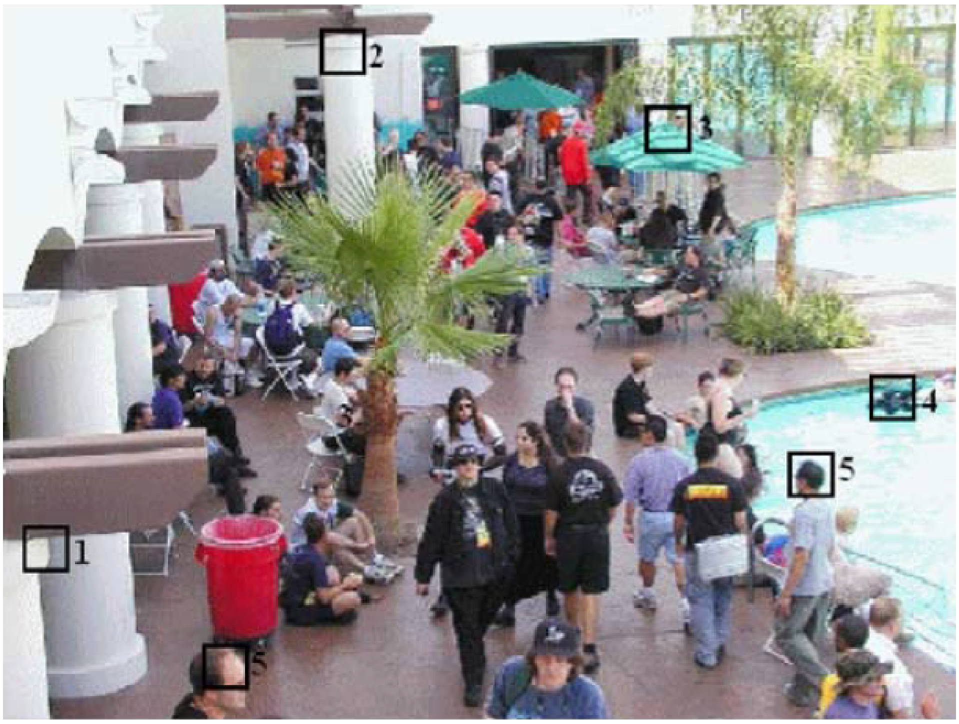


Figure 6: *Example password of the PassPoints scheme. [19]*

2.6 Adapting Established Approaches to Virtual Reality

The publication ”Seamless and Secure VR: Adapting and Evaluating Established Authentication Systems for Virtual Reality” by George et al. evaluates minor adaptations that had to be made in order to use two already established authentication schemes in VR. [31]

George et al. implemented both the PIN scheme and the Android Pattern scheme in VR for the HTC Vive HMD. In their study they observed how likely an attack is successful on both schemes assuming an outside attacker who is able to fully observe hand and controller movement of the victim. The attacker was not able to see the victim’s view inside the HMD. Additionally they changed variables like surface size, large versus small pointer etc. to observe how these changes would influence both usability and security of the authentication scheme. During their study they also collected information like login durations and the success rate of an outside attacker. [31]

Overall this study showed that both the PIN scheme and the Android Pattern authentication translate decently well into VR and offer a reasonably secure solution to authenticate oneself. Both schemes were transfered directly with no changes whatsoever in their core functionality and their measured performance matched the data that has been collected in real-world experiments using the same authentication schemes. [31]

This evaluation will be one of my core sources during the study section and I will compare the scheme I am developing to both the PIN and Android Pattern scheme which will be implemented just like this paper explained them.

3 Concept

In this chapter I will explain the ideas that went into the concept of my newly designed authentication scheme.

3.1 Foundation

During my research I noticed, that the amount of authentication method publications specifically targeting VR or AR devices was very limited. After extensive study about authentication methods in general it was clear to me that a graphical authentication scheme is the best option for a new authentication scheme in VR or AR when both security and memorability are critical. While biometric authentication is an easy and secure approach for the user, it has the significant downside that biometric passwords can not be changed once compromised. As a consequence, one should really think twice about using biometric authentication everywhere. I wanted my scheme to be usable without using any biometric information for this particular reason.

A graphical password is in theory superior to knowledge-based authentication methods like alphanumeric passwords or PINs in many ways, regardless of the device you authenticate on. This is mainly because the precise recall of an alphanumeric code is much harder for the human brain than the recognition of previously already known pictures. [13, 16, 28]

While graphical authentication methods should be superior to knowledge-based alternatives, it is quite surprising that there is still no established graphical authentication scheme. There are a lot of different approaches, but none of these has been able to establish itself. [14, 16]

I believe that this is mainly because of the flexibility a knowledge-based system like PIN-authentication offers. There are no real specific hardware or software constraints, one just needs any kind of hardware to input a series of numbers or letters. Graphical passwords on the other hands usually require much more advanced hardware. Most of them rely on a sufficiently large high resolution screen, such that the user is able to properly see details on it. Many authentication methods require this screen to additionally support colors and some form of cursor to e.g. select one out of multiple pictures.

While at first glance these requirements do not sound very strict, in practice there are a lot of scenarios in which people constantly authenticate, without these requirements being met. The most obvious example is the regular Electronic Cash and Credit Card system. Almost all of these authentications use a simple four digit PIN code to authenticate the user. The hardware is already in place and established, so almost everyone today is familiar with the PIN scheme. Changing these deeply integrated systems in daily life might take a very long time to adapt to, but there are good reasons for it. I believe that the advantages to use a graphical authentication scheme especially in VR and AR over e.g. the PIN-scheme are tremendous, which is why I decided to create a graphical authentication scheme which is specifically tailored for usage in the VR or AR.

The existing graphical authentication schemes can be split into two basic categories: **recall-based** and **recognition-based**. In an extensive study on a wide range of graphical authentication schemes, it was discovered that recognition-based authentication schemes are usually more complex which results in longer authentication durations. On the other hand recall-based schemes had the main disadvantage of users struggling to remember the precise sequence that was necessary to authenticate. [14]

With my newly developed scheme I will not strictly enforce the user to use either recognition or recall. Depending on the picture the user decides to use with my authentication scheme it will be either recognition or recall. Generally the user will be encouraged to use a personal picture to authenticate which would make the scheme a recognition-based authentication scheme.

3.2 Overview

My newly developed scheme called "Personal Environment Authentication" (PEA) will authenticate the user in a virtual world which consists of a personal equirectangular image. An equirectangular image is essentially a 360-degree capture of the cameras' surroundings. The "password" now consists of a series of clicks on objects in this image which have to be performed in the correct order.

In the registration phase the user is able to define exactly which parts of the image have to be clicked on to be considered a correct hit and their order. During the login the user just has to click on the defined areas in the set order to be successfully authenticated. Any deviation from the set order, or a click which missed the previously defined area will lead to an internal reset of the scheme. This means that the person who wants to authenticate has to start over by clicking on the first object again to 'restart' the sequence. It is crucial that this reset only happens internally and is not visible to the user, as an attacker could use this information to perform an incremental guessing attack on the password.

Example: I use a 360-degree image of my living room as background in the PEA scheme. In my living room I have a lot of objects placed on shelves, my closet and the two desks which are standing in my living room. There are also many pictures from vacations on the wall. I can now click on gifts from my friends in the chronological order I received them and use this sequence of objects as my password.

It is quite clear that for a stranger it will be impossible to forge this password except for starting a brute-force guessing attack on the entire password space. When using a picture which contains a sufficient amount of details and the number of objects used in the "password" is reasonably high, this guessing attack will be almost impossible due to the pure number of possible combinations[6]. The attack becomes even harder because an attacker has no way to distinguish a 'correct click' from a 'false click', avoiding incremental brute-force attacks (forging object after object).

The number of such possible object sequences with a specific order used as passwords is almost endless. One could use events in the past or any story one could think of. The more personal this story is, the harder it is for an attacker to systematically guess the password and the easier it becomes for the user to remember it during the login process.

In fact it is easy to remember a personal story associated with a specific environment even after a long period of time. This is especially true because my scheme will display the user her selected environment which will assist her in recognizing it and remember the story associated with this picture/environment.

This effect can be replicated, e.g. by looking at pictures of past vacations, or any picture that got taken in a context one was involved in. In many cases one will associate a specific story with them which is not apparent for an outsider by just looking at the picture. Those are the perfect images to use in my authentication scheme, one just has to find a sequence of clicks on objects in the image which represent this particular story.

3.3 Design

The design of my PEA scheme is minimalistic and as simple at possible. Displaying a lot of information in the user interface can reduce the immersion significantly [1], as it would cover parts of the 360-degree panorama surrounding the user. The goal is to make the user feel as if she currently is inside her chosen environment, so any additions which do not exist in the picture should be avoided wherever possible. [1]

A click by the user on the panorama performed during registration will spawn a green sphere with its center of mass on the coordinate the user clicked on in the panorama picture. The sphere defines the exact area in which the user has to click on during the login to be considered a 'correct click'. A confirmation of this sphere will change its color

[6] As discovered during my study, four objects might already be sufficient to make a guessing attack almost impossible, as shown in section 6.3

to brown, such that it is easy to differentiate a newly placed sphere (which can still be resized etc.) from an already confirmed sphere.

In a login session, these spheres will exist but their rendering is disabled so that the user can not see the areas in which she has to click. To authenticate successfully the user has to click inside these invisible spheres in the same order she created them during registration. The only feedback that is offered in this phase is the spawning of brown spheres on every click the user performs. These orbs offer no functionality at all, they are only displayed for visual feedback regarding the registration and precision of user input. This feedback offers the possibility to check what was already clicked on and also assists the user in identifying an accidental or imprecise click.

During pilot experiments I noticed that displaying the current status of the application is crucial (as some users e.g. did not know if they are in a registration or login session) and that some users preferred to avoid swipe gestures whenever possible. For this reason I implemented a minimalistic User Interface (UI), as shown in figure 7 which is static and fixed to the camera (which represents the user's field of view in Unity). It displays the current status of the program, which can be "user selection", "registration" or "login" and buttons which act as alternatives to the swiping gestures, which are used to undo and confirm actions.

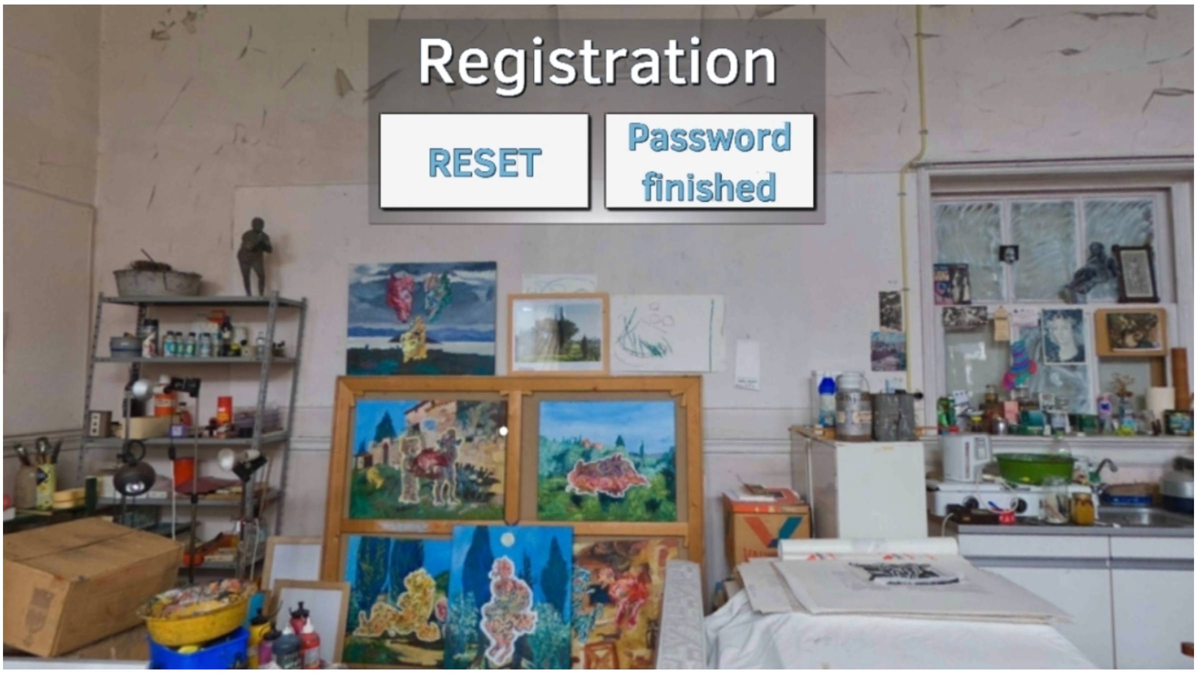


Figure 7: *User Interface of the PEA scheme. [32]*

For safety reasons I disabled the feature to save the password using just a swipe to the right. This functionality can only be triggered using the "save password" button on the UI. This was done to minimize the risk of an accidental password confirmation. Additionally every input has to be confirmed separately to make sure that the user is able to notice mistakes in time, before saving the password. The background of the UI is a glass-like material which the user is able to see through. The material is reflecting which raises it from the panorama without it being distracting.

3.4 Interactions

To make use of the increased security multi-modal approaches offer, as explained in my related work chapter [25] my novel environment scheme will make use of 1) Head Gaze, 2) Controller Movement and 3) touchpad interaction at the same time to interact with the application.

Head tracking will translate one-to-one into the virtual environment, which is taken care of by functionality the GoogleVR SDK already offers. This form of interaction with the three-dimensional scene feels natural and is therefore easy to use and learn. [5] A one-to-one mapping of movement from the real world into the virtual world also increases the immersion of the application significantly. [5]

Controller movement will similarly translate into the virtual environment using functionality provided by the GoogleVR SDK. The controller will act as a laser pointer with which the user is able to point on objects and interact with them using the touchpad of the Google Daydream Controller. To increase the immersion of this interaction, the user will be able to see the controller in the virtual world just like she would in the real world. This additional feedback is important, such that the user is able to learn these types of interaction quickly, recognizing mistakes on her own (e.g. accidentally tilting the controller).

The same graphical feedback for the controller's position and rotation is also used to provide feedback regarding the interactions with the touchpad of the Google Daydream controller by visualizing any thumb movements of the user including the triggered gestures. This is explained in detail in chapter 4.2. The interactions with the touchpad are limited to swiping gestures and a simple press because most users are already familiar with these movements from using a touchscreen on a smart-phone or similar device.

4 Implementation

The following chapter will explain the different measures and solutions I came up with while developing my novel authentication scheme including an overview of the resulting architecture design. I will provide detailed description of problems I discovered and the solutions I came up with to solve these. Additionally the **System Overview** section will define the exact hardware and software setup I was using during development, testing and evaluation during the user-study.

4.1 System Overview

Software

The core Frameworks I was using to implement my authentication scheme was Unity Engine 2017.3.0f3 for both macOS 10.12.6 and Windows 10 (Version 1511). To interact with the Google Daydream System the official Google VR SDK for unity (release 1.70.0) was used which supports both Google Daydream and Google Cardboard. The Google VR SDK was especially helpful to implement integration of the HMD and the Daydream Controller into the Unity scene.

Hardware

I decided to use Google Daydream (as shown in figure 8) combined with a Google Pixel (Generation 1) (as shown in figure 9) as the VR HMD because it offers a reasonable resolution for a very low price and because the GoogleVR SDK is very well documented and easy to work with.

Figure 8: *Picture of Google Daydream View. [33]*

Figure 9: *Picture of the Google Pixel (Gen 1). [34]*

As an input method the Google Daydream Controller was used which is included with the Google Daydream kit. For my application I mainly used the touchpad referenced as 1 in Figure 10, because I did not want to limit my scheme to a specific number or layout of buttons. Usage of a touchpad seemed very reasonable because all the original controllers of the most popular HMDs feature a touchpad (GearVR, Daydream, Oculus Rift, HTC Vive, etc.).

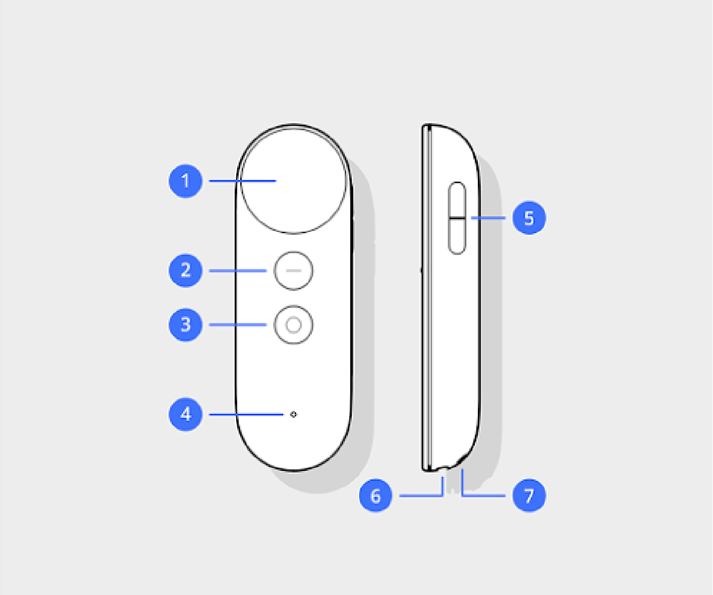


Figure 10: *Interface of the Google Daydream Controller. [35]*

4.2 Implementation

Virtual Reality Worlds using Equirectangular Images

To make use of the increased immersion and the additional dimension in general, it was essential to make the user feel as if she was in the visualized environment. So I essentially used 360-degree panoramas in the unity scene as "background". These panoramas are suggested to be pictures which are personal to the user.

The easiest way I found to use captured images of 360-degree spherical "panoramas" in my authentication scheme was to use an equirectangular image and changing its texture shape in Unity to 'cube' and its mapping to 'Latitude Longitude Layout'. This enables selection of this picture as material for the skybox in the Unity scene. Equirectangular images are images of two by one ratio which display a 360 degree horizontal field of view and a 180 degree vertical field of view in a single stitched image. The format of an equirectangular image is standardized for all manufacturers of spherical cameras. [36]

I decided to use equirectangular images as opposed to a cubic format mainly because of equirectangular images' popularity which makes it easier to demonstrate my scheme on a wide range of different pictures.

A large collection of equirectangular pictures can be found on the equirectangular flickr group[7]. All these pictures can be used as a skybox in my authentication scheme.

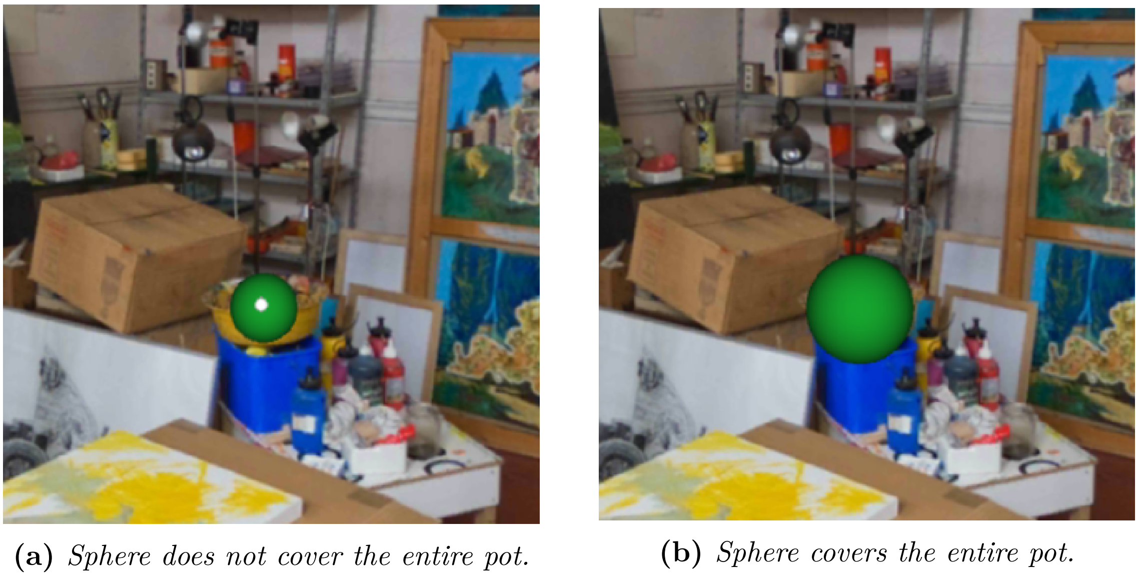
For equirectangular images of sufficient resolution, this approach works really well and gives a feeling of depth to the user which makes it feel as if one is currently standing in the room in which the panorama was taken.

Rescaling of Spheres

If there is a large object in the panorama picture, the user would like to use as part of his password, the default sized sphere might not entirely cover the selected object as it is the case in figure 11(a). Therefore the possibility of resizing the spheres, which define the area the user has to click on during the login phase for it to be considered a 'correct click', is very important to achieve maximum flexibility for passwords.

Otherwise the problem could occur that logically correct click (clicking on the object which was meant to be part of the password, e.g. clicking on the edge of the yellow pot in the example figure) is rejected by the system because the sphere did not sufficiently cover the large object. Remembering not only the object, but also where exactly a user has to click on this object can be quite hard and is unnecessary memory load that can be avoided entirely by allowing the user to flexibly scale the sphere to the desired size.

[7] `www.flickr.com/groups/equirectangular/`



(a) *Sphere does not cover the entire pot.* (b) *Sphere covers the entire pot.*

Figure 11: *Rescaling before and after. [32]*

My first approach to rescale spheres in the registration phase was inspired by window rescaling in most operating systems. The user has to click at the edge of the sphere and then move the cursor away from the coordinate she clicked on, which would result in the sphere growing further the more the cursor is moved away from the coordinate she clicked on. The user could decrease the size of the sphere by clicking on it and then moving the cursor further towards the center of gravity of this sphere.

Unfortunately I underestimated some drawbacks this solution introduced. The two most prominent ones being:

- Often times users misplaced their sphere just a little bit. To correct this they had to place the sphere somewhere else first just to be able to click again very close to the previously misplaced sphere, because a click inside the sphere would trigger the rescaling process instead of moving the sphere to the new click. This is unintuitive, and it would be desirable to have the spheres 'click-through' to allow for minor position changes by clicking through the sphere to a coordinate close to the current center of mass.

- During testing on early beta versions of my application, I discovered that a significant amount of testers expected spheres to be confirmed by double clicking on them. Instead, this would start the rescaling process, which often led to confusion.

I solved these problems by completely redesigning the rescaling process of spheres using the swipe menu explained in the next paragraphs.

Swipe Gestures

Due to the problems the first rescaling implementation introduced, I decided to use swipe gestures to control the rescaling instead of the window rescaling approach. Fortunately Google Daydream Elements, which is available on GitHub, contains a prefab for a Swipe Menu.[8][9]

This swipe menu enables user-interaction using swipe gestures while still maintaining the core functionality of the controller with its point-and-click approach. The decision to resize spheres using swipe gestures also significantly reduced the amount of controller movement necessary when compared to my previous window rescaling approach. An alternative solution using UI buttons to resize spheres, would have been even more physically demanding in terms of additional movement and would significantly reduce the immersion as larger parts of the picture would be covered by UI elements.

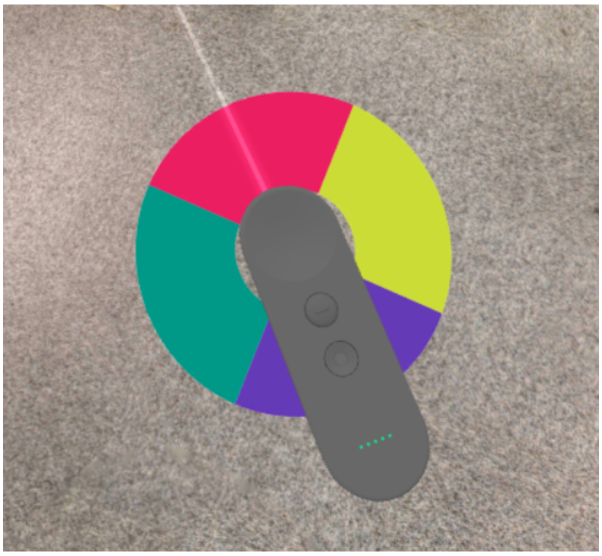


Figure 12: *Addition of Swipe Menu to Daydream Controller in VR.*

The swipe menu also offers visual feedback to swipe gestures on the Google Daydream Controller. This feedback was especially useful during the tutorial session of the study to practice working with the touchpad. As one can see in figure 12 four colored shapes are displayed around the touchpad and when e.g. a swipe to the left is triggered the green shape will move to the outside, indicating correct detection of the gesture.

Rewriting parts of the SwipeMenu class enabled me to forward frames in which a swipe was detected to my 'TouchPadObserver' class. Due to the inconsistent frame-rate of my

[8] `developers.google.com/vr/elements/swipe-menu`

[9] `github.com/googlevr/daydream-elements`

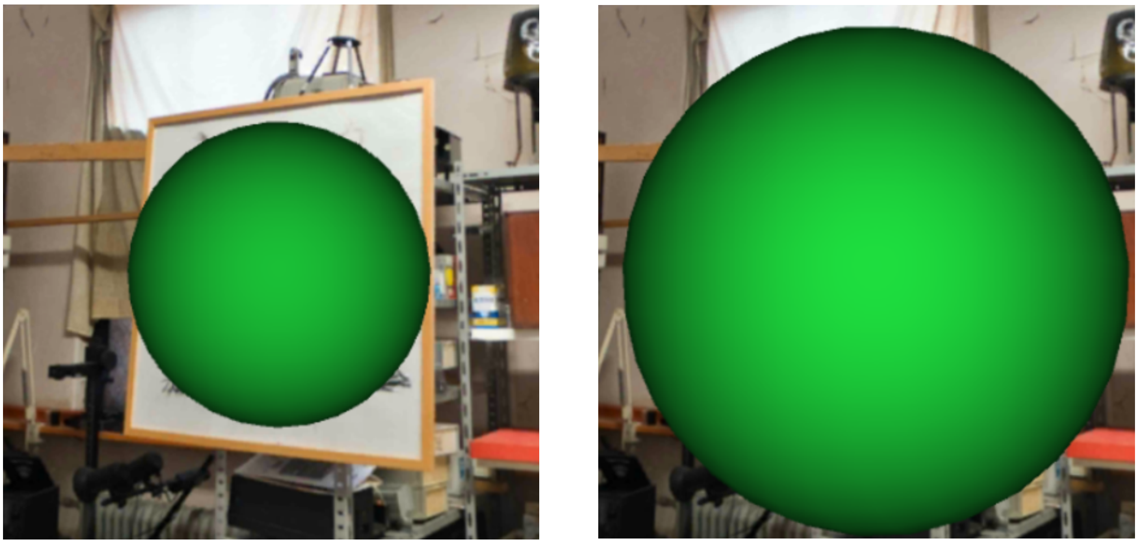
application (running on the smartphone), I had to differentiate between frames indicating the start or continuation of a swipe gesture using real-time-thresholds. This is essential for e.g. the 'back'-feature to work as intended because otherwise swiping left would be detected on multiple frames by the Google Swipe Menu triggering the 'back'-feature repeatedly.

The size of a sphere is increases by 20% of its default size by swiping upwards while swiping down decreases the size by 20%. Further testing discovered, that users wanted to increase the sphere size by more than 200% in some scenarios. Swiping up ten times in a row led to frustration, so I decided to extend the swipe gestures by implementing a special kind of gesture called 'held-swipe gestures'.

This specific gesture is triggered if the swiping finger does not leave the touchpad but stays there for at least 0.7 seconds. Now every 0.7 seconds a new trigger for a swipe will be sent without having to move your finger. This feature has proven to be a huge usability upgrade when rescaling circles in my authentication scheme compared to previously implemented methods mentioned in the *Rescaling of Spheres* section above.

Polygon Mesh Creation

While an arbitrary scaled sphere is flexible and can be used to cover any shape, it would be advantageous to be able to adapt the shape beyond just circular spheres. This would significantly enhance security because e.g. covering a rectangle using a scaled sphere either includes a large amount of pixels outside of the form as shown in figure 13(b) or does not cover the rectangle in its entirety as shown in figure 13(a). Both would lead to 'logically false' behaviors of the system not accepting clicks inside the rectangle or accepting clicks outside of it.



(a) *Does not cover the entire drawing* **(b)** *Covers more than just the drawing*

Figure 13: *The problems of spherical shapes. [32]*

To solve this very prominent issue in testing, I decided to implement the creation of custom shapes. The `meshBuilder` class constructs meshes consisting out of triangular shapes which can be created by the user. The custom meshes are created by consecutively creating spheres in clockwise order. As soon as the user created at least three consecutive spheres, the mesh will be generated by connecting the three consecutively clicked coordinates. The custom shape will displayed live to support the user in extending it using further clicks performed in clockwise rotation. There are no limitations to size or complexity of a custom shape.

These custom shapes solve the problems displayed in figure 13 entirely as shown in figure 14 and can be used to cover arbitrary complex shapes.

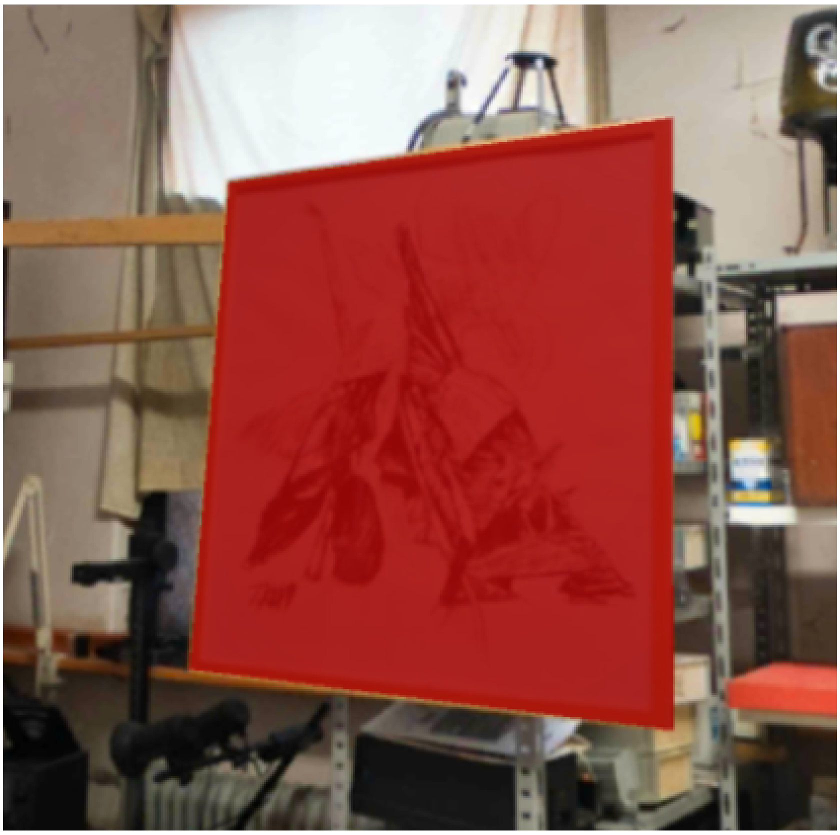


Figure 14: *User created custom shape using four clicks to cover the drawing. [32]*

Except for the creation process, which differs from regular spheres, the custom shapes behave just like circular spheres. They can be reverted with a swipe to the left and confirmed with a swipe to the right.

Study Limitations

To decrease the risk of user-caused ’errors’ occurring during the study, I disabled some features of my authentication scheme. The most obvious limitation was to limit the user-selection to just a single user and have every participant authenticate with the same user. This avoids that a participant registers as user01 but tries to login as user02 on accident.

Another important limitation was to restrict the functionality of the ’back’-function which is triggered by swiping left. During the study it was e.g. not possible to swipe in the login section, when no spheres are active to go back to the user selection. This was essential to make sure, that participants do not accidentally leave the experiment in which the logging etc. takes place.

The last limitation I decided was to disable all but a single panorama skybox for my authentication scheme. This was done for multiple reasons. The most important one being that it was much easier for participants to get familiar with a single picture compared to multiple pictures. Additionally the usage of different panoramas would have made it nearly impossible to draw conclusions regarding the security of my scheme, as the security of different panorama skyboxes are inherently different, mainly because of the different object densities in the pictures.

For real-world usage all these limitations are not necessary, because an experienced user will have no trouble dealing and making use of these features. Especially using different skyboxes will increase the security of the authentication scheme significantly, as this is another parameter the attacker would have to ’guess’ to authenticate successfully.

4.3 Architecture

My authentication scheme mainly consists out of six core classes interacting with each other. In the following I will explain the core functionality of these classes and provide an overview of interactions between them.

Room

The room script runs concurrently to all other functionality in my authentication scheme and is attached to a large invisible sphere which surrounds the user. Inversion of the surface normals of this sphere causes it to register clicks on the inside. It is hit whenever the user clicks without hitting either an interface element, an existing sphere or an existing custom shape. It is essential to handle these clicks separately, because they reset the password progress in the login step or create new shapes in the registration process.

SwipeMenu

This class was provided by Google[10] [11] and detects frames in which a swipe gesture was detected. I rewrote this class to forward detected gestures to my TouchPadObserverClass which takes care of interpreting the received gestures.

TouchPadObserver

The TouchPadObserver class interprets the detected SwipeMenu gestures by checking duration and direction of the last detected swipe. Depending on that gestures will be interpreted differently. This class is especially important to allow for controlled single swipes (does not detect two swipes) while also allowing swipe gestures to be 'held' which means that the user can swipe up and hold his finger there to grow a sphere in the scene continuously as long as her finger is held down on the touchpad. Additionally the functionality of the 'back'-gesture (swipe to the left) is implemented in this class.

CubeTeleporter

The CubeTeleporter class is attached to every sphere during registration and login. It implements the growing and positioning functionality of spheres and takes care of detecting 'correct clicks' in the login phase of my application.

[10] developers.google.com/vr/elements/swipe-menu

[11] github.com/googlevr/daydream-elements

buildMesh

This class takes care of the same functionality as the CubeTeleporter class, except for custom shapes. For that reason it needs a lot of additional features like the calculation of meshes and the tracking of vertices and triangles of this particular shape. Additionally every custom shape has the functionality to create a non-intractable copy of itself, which is necessary to track and display previously created custom shapes in the registration process correctly.

LoginInManger

The LoginInManger class is the controller class of my authentication scheme and is running concurrently throughout the entire runtime. It coordinates the functionality of all the previously explained classes and controls the entire unity scene and control flow which is split into three main states being: user selection, registration and login. Depending on the current state, different UI elements have to be displayed and saved passwords have to be loaded from memory depending on the selected user. The LogIn-Manager class also tracks the temporary password while a registration takes place, resets the password progress on an incorrect click in the room during the login phase, saves a password to memory after registration has been completed etc. Additionally, it implements the logging of information during the registration and login phase which was used in the evaluation of my user study. The LoginInManger also makes sure to disable the rendering of the spheres and all custom shapes before starting a login process, such that the scheme does not leak any information to a potential attacker.

4.4 Additional Security Enhancements

A very obvious and important improvement to security, already implemented in early iterations of my software, was a random horizontal rotation of the panorama which occurs at the very start of every login. This random rotation is important to decrease the risk of a shoulder surfing attack significantly, because without it an attacker simply has to replicate the exact movement the user who authenticated successfully performed. While this would still not be trivial, it is a very feasible attack, especially against shorter passwords without this rotation.

The random rotation can be performed, before the panorama has been loaded at the very start of the authentication process, so it does not introduce any discomfort because it is not visible to the user.

A feature which was disabled during the study of the PEA scheme (already mentioned in section *Study Limitations*) is the possibility to change the panorama background using swipe gestures up and down. This adds an additional layer of security, because an attacker would not only have to be able to find the correct objects in a picture but also the correct panorama picture itself first. Because the login section starts with a randomly chosen panorama, the correct panorama is not leaked by just counting the amount of swipes up/down the user performs in his login or registration. Enabling this feature will increase the security of the PEA scheme significantly compared to the results discovered during the conducted user study.

5 User Study

5.1 Goals of the Study

The goal of the study conducted as part of my thesis was to compare the scheme I created with two of the most prominent existing solutions, being the 'Android Pattern Authentication' in which the user draws a secret pattern on the screen and classic ten-digit 'PIN' authentication, in terms of security and usability. The experiment gathered as much information about effectiveness, efficiency, safety and usability as possible while also observing the learning behavior of novice users. Due to the time limitations of a bachelor thesis I was not able to compare my scheme to existing solutions regarding long-term memorability of multiple passwords.

5.2 Participants

I recruited thirteen volunteer participants, with two of them being female and eleven of them being male. Unfortunately I had to exclude one participant from my study because of a language barrier. The twelve valid participants' age ranged from 21 to 27 years with mean age being 23.5. Six of the participants usually wear glasses or contact lenses, but none of them had to wear their glasses inside Google Daydream.

Out of the twelve participants three already authenticated themselves in the VR at least once using a virtual keyboard implementation which they overall described as a very unpleasant and slow experience.

All participants classified themselves as familiar with both the PIN and the Android Pattern Authentication (e.g. used on their smartphones in the past). Although they classified themselves as familiar with the Android Pattern scheme, two of the twelve participants did not feel comfortable with this authentication method. This was not the case for the PIN scheme. None of the participants had any experience with the PEA scheme prior to the experiment.

In addition to the participants, I recruited a male attacker with VR experience who was trained extensively in all three authentication schemes. He was not only trained at using the schemes himself, but also observing users who authenticate themselves. During this process, the attacker was allowed and encouraged to take notes. After observing the authentication of users, the attacker practiced attacking their password by trying to replicate the movement performed by the authenticating person. After multiple hours of training, the attacker developed different strategies to attack each of the authentication schemes which will be explained in detail in chapter 6.

5.3 Experiment Setup

The Unity project ran on a first generation Google Pixel, which was the same device that was also used during development phase. The smartphone features a five inch AMLOD screen with a resolution of 1080 × 1920 pixels resulting in a pixel density of 441 pixels per inch[12]. The pixel was placed in a first generation Google Daydream device offering a 90 degree field of view. To control movement in VR the accelerator and gyro-meter are used. Additionally the user held the Daydream Controller, which was used to interact with the virtual environment, in his preferred hand.

The Unity application was developed using Unity 2017.2.1f1 on macOS and the official GoogleVR SDK (release 1.70.0). The smartphone inside is required to run at least Android Nougat 7.1. For our experiments I used Android 8.1.0.

To cast the screen of the Google Pixel to a TV I used a first generation Google Chromecast, which was plugged into a TV. Using the Google Home Application for Android I was able to mirror the Google Pixel's screen onto the TV. For some parts of the experiment a beamer was used instead of a TV.

The live video-capture and display of the smartphone screen which was mounted inside Google Daydream was mainly used to allow for better instructions during the tutorial phases. It was much easier for myself (the instructor) to recognize and understand the problems the participant was facing, with the ability to see exactly what the participant is seeing inside the HMD.

Additionally to the live video feed, a Panasonic HDC-SD300 full HD camcorder mounted on a tripod was used to film the participant during specific parts of the experiment (the login sessions), such that these parts of the experiment can be watched by the attacker afterwards.

The application running on the smartphone which is mounted inside Google Daydream captured various data throughout the entire experiment. I primarily used this form of information as a source for the evaluation process because it is precise, detailed and objective data that can be compared for all participants to evaluate the participants' performance in the three different authentication schemes.

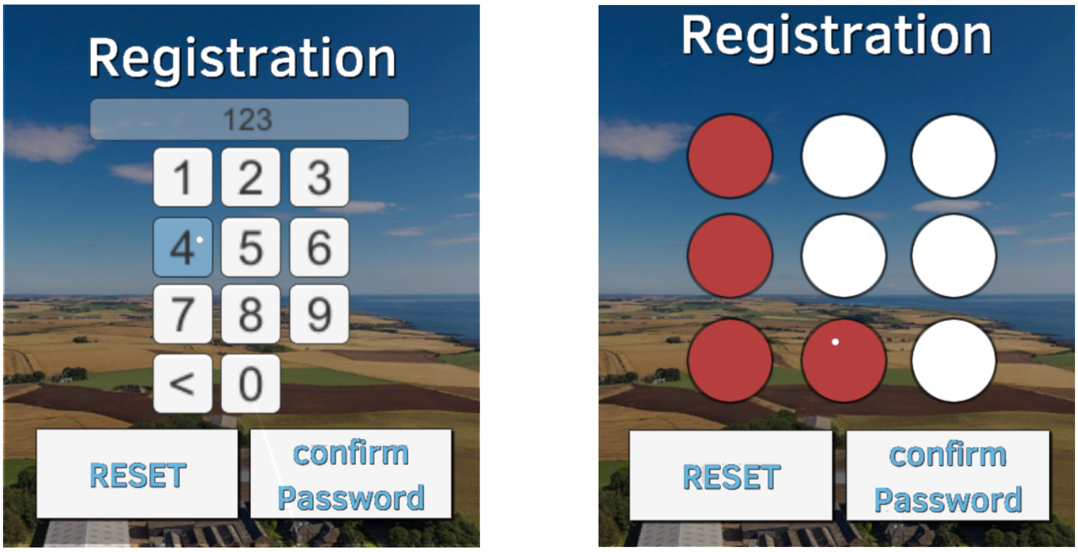
The experiment took place in a 'poor mans usability lab' in which I was observing the mirrored screen of the smartphone which is being put into Google Daydream. I made sure that all participants have sufficient space around them to turn and move their arms freely without the danger of hurting themselves or anyone else. To further reduce the risk every participant sat down on a rotating office chair throughout the whole experiment. The usage of a usability lab ensured that the chance of interruptions from

[12] `store.google.com/product/pixel_phone_specs`

outside are minimized creating ideal conditions to observe the participants' behavior inside the different authentication schemes.

5.4 Study Design

Independent Variables: The study followed a multi-level design in which the experiment phases are repeated with *three different authentication schemes* for each participant. The different authentication schemes are: PIN authentication, Android Pattern authentication and PEA. In the well known PIN scheme, a user authenticates with a pass-code which consists of a fixed number of digits ranging from zero to nine. The pass-code is entered using a classic telephone keypad as shown in figure 15(a). Another fairly well known authentication scheme, called Android Pattern, authenticates the user with a gesture drawing over a three by three grid. The order in which the user hovers over the parts of the grid represents the password as shown in figure 15(b). The third authentication scheme, which was used in the experiment is the novel PEA scheme as it was explained in sections 3 and 4 of this thesis.



(a) *PIN scheme* **(b)** *Android Pattern scheme*

Figure 15: *Screenshots of my VR implementation of PIN and Android Pattern scheme.*

It is especially important to counterbalance the independent variable *authentication scheme* as explained in the following counterbalancing paragraph.

Another independent variable was the *predetermined password-length* that was used by the user to register or login with in each of the schemes. I selected password lengths of two, four and six to be inside the most prominently used range for most passwords, which is the reason why I used these password lengths during the experiment.

Participants first had to register and login using a password of length two. In the next iteration a password of length four and in the last iteration a password of length six was used. This sequence repeated itself for all three authentication schemes.

All passwords were randomly generated using python scripts developed by me. For the environment scheme I only used a pool of the 45 most unique objects (handpicked by me) in the picture to simplify the learning process of users. This could easily be extended as there are a lot more visible details in the picture. To simplify the creation of Android Pattern passwords I used a list created by the GitHub user `delight-im` which contains all possible Android Pattern passwords up to a length of nine[13]. My script randomly selected a pattern of correct length from this list using the in python `v3.1.5` built-in pseudo random generator. The same pseudo random generator was also used to generate a random PIN password.

Counterbalancing of Independent Variables: It seemed likely that learning or practice effects will occur especially in an environment in which many participants might be unfamiliar with the device and software they are using. For this reason it was absolutely crucial to apply counterbalancing and change the authentication method variable using a Latin Square [37]. The application of a Latin Square ensured that the order of the different schemes will be mixed such that every authentication scheme was used as first, second and third scheme the same number of times throughout the whole experiment.

[13] `github.com/delight-im/AndroidPatternLock`

Dependent Variables: The dependent variables of this study were captured using the logging which was implemented in all three authentication schemes to ensure a fair and objective comparison in the following evaluation.

I tracked the *time it took to successfully perform a registration* in the unity implementation. This timer starts with the click on the correct user in the main menu of the respective application and finishes with the successful completion of the entire registration process.

The *time it took to login* was tracked in exactly the same way starting with the click on the correct user and finished with successful login. In this case it was possible that the participants are not able to login successfully on the first try. The timer was not restarted in such a case and ran until the login was completed successfully.

I also tracked the *number of "UNDO"* which have been performed by the participant. This undo feature is implemented in all schemes in exactly the same way: It simply reverts the last "password digit' (or sphere/custom shape in the PEA scheme) selected by the user. This variable indicated how many times the participant miss-clicked in the experiment but was able to notice his mistake before submitting the password.

To track the number of unnoticed mistakes I also logged the *number of failed login attempts*. A login was considered failed, when the application did not indicate it as being successful. This most likely indicated an unnoticed miss-click or inaccuracy during the login process. Another reason for a login to fail is when the participant forgot parts of the password he registered with. This did not occur for any participant in the experiment I conducted.

Using the captured video during the login process of a participant an expert attacker had a limited amount of time to perform a strategical guessing attack on the created password. For this the following information was logged by the application:

1. *If the attack was successful in the given time.*
2. *How many seconds it took the attacker to perform the attack.*

5.5 Procedure

After welcoming the participant and going through the consent form, which explained possible risks of this study and what kind of personal data is being captured, every participant was given an introduction to the hardware which was used. Special focus was put on the Daydream Controller and how it is being used when interacting with the virtual environment. After clearing up any questions the participant had, she was assisted in fitting the headband of the Google Daydream headset, such that the user was comfortable and the HMD was fitted securely.

Afterwards the user was being tutored in the first authentication scheme (determined by balanced order as explained above). For this reason a special tutorial was implemented for all three authentication schemes, which was used to explore all the different features this particular authentication scheme has to offer. This tutorial was special in the way that it could be repeated as many times as desired, such that the user was able to experiment with it as much as she wanted or needed.

I followed a strict guideline during these tutorials to ensure that every participant achieved a comparable amount of experience in these tutorial sessions. Every user experimented with all the features the particular authentication scheme has to offer.

Whenever the user felt comfortable with the particular authentication scheme and all questions have been answered the experiment started. It always followed the same structure: I communicated a randomly generated password for the current authentication scheme to the participant. Afterwards the participant registered with this password (questions and communication in this step was allowed) and was given as much time as she needed to remember the password. In the last step the participant performed a login with the password she had previously registered with. Only this last part of the experiment (login step) was being filmed using a camcorder and there was no communication between instructor and participant allowed in this step of the experiment.

For every authentication scheme this experiment pattern was repeated for randomly generated passwords of size two, four and six. An additional help, offered to every participant, was a sheet of paper on which the password was written on, in numeric and graphic form. This was especially useful for the Android Pattern authentication scheme, as it was easier for some people to remember the pattern instead of the 'numeric code' it represented. The graphic form of the Android Pattern scheme included a drawing of the movement on the three by three grid using a filled circle as start indication and an empty circle as end indication. Most of the participants made use of the possibility to look at the written passwords on a sheet of paper.

After successfully performing the experiment for all three authentication schemes, the user were given a User Experience Questionnaire (UEQ) [38] and a NASA-TLX questionnaire for all three authentication schemes as well as a single demographic questionnaire, developed by myself. All three questionnaires can be found in the appendices of this thesis.

5.6 Attacker Procedure

Overall the attacker had a maximum time of five minutes to perform an attack on a single login one of the participants performed.

Before these five minutes, the attacker was given as much time as he needed to prepare whatever he deemed useful to perform his attack afterwards.

When the five minutes started, the attacker first watched the captured video of the login procedure by the participant two times at regular speed. After he finished his notes he was allowed to start his first attack using the Unity application, which was allowed to take at most 60 seconds. During this time the attacker had no limitations at all, except for the time frame. This means that he could try as many passwords as he wanted until he either entered the correct password or until 60 seconds passed.

If he failed the attack he had the remainder of the five minutes to watch the video as many times as he wanted. Additionally he was now allowed to pause and slow the video down at his desire. Whenever he wanted the attacker was allowed to start a second attack in my application, which again took at most 60 seconds. An attack was considered successful when the attacker was able to authenticate successfully with the password the participant created. There were no attacks allowed beyond this second attack, even if the five minutes were not entirely used.

I decided to use this two-phase attack design, to simulate both a 'casual attacker' and also an 'expert attacker' separately. The expert attacker has both hardware and software assistance, while the casual attacker, just observes the login. To achieve a realistic scenario the pool of 45 objects, which was used to randomly generate passwords in the PEA scheme, was not given to the attacker.

5.7 Hypothesis

- The PEA method is more secure than the PIN and Android Pattern authentication scheme against shoulder surfing attacks in a VR environment.
- The PEA method offers improved usability compared to the Android Pattern Authentication or the PIN Scheme in a VR environment.

6 Results

The following section will summarize the results that were gathered during my user study.

6.1 Registration and Login Time

The registration time is the duration it takes a participant to register with the randomly generated password she is told (in seconds). It is measured in exactly the same way for all three different schemes and describes the elapsed time between the completion of the user selection and the press on the 'confirm password' button.

The login time describes the time it takes an participant to login with the randomly generated password she previously registered with. It is also measured in exactly the same way for all three different schemes starting with the completion of the user selection and finished with the successful authentication. An authentication attempt which fails (e.g. due to incorrect password) does not finish the login timer. The participant has to retry until she authenticates successfully or until the instructor interrupts her.

The PIN authentication scheme offered the fastest registration time with a mean of $7.79s$ ($SD = 4.04$) shortly followed by the Android Pattern scheme with a mean of $9.30s$ ($SD = 5.00$). Authenticating with the password, the participant previously registered with, was the fastest in the Android Pattern scheme with a mean of $4.16s$ ($SD = 1.56$) shortly followed this time by the PIN scheme with $5.39s$ ($SD = 1.67$).

In figure 16 below it is very obvious, that the PEA scheme is by far the slowest for both registration, which took on average $47.32s$ ($SD = 23.62$), and login process, which took $16.73s$ ($SD = 13.04$) on average in my experiment.

A univariate ANOVA shows, that there is a statistically significant difference for the authentication schemes ($p < 0.01, F_{(2,213)} = 116.279, \eta^2 = 0.405$). For the group of Login and Registration the univariate ANOVA also found a significant effect ($p < 0.01, F_{(1,214)} = 27.687, \eta^2 = 0.115$). Additionally the ANOVA found a significant interaction between registration/login time and the authentication scheme and the registration/login group with $p < 0.01$ ($p < 0.01, F_{(2,210)} = 32.772, \eta^2 = 0.238$).

Looking at the login and registration times for the different password lengths in figure 17 it shows that the mean login time of the PEA scheme of a password size two with $8.19s$ ($SD = 2.33$) is not that much higher than the mean login time of the PIN and Android Pattern scheme with a password size of six with $6.83s$ ($SD = 1.11$) and $5.83s$ ($SD = 0.93$) respectively.

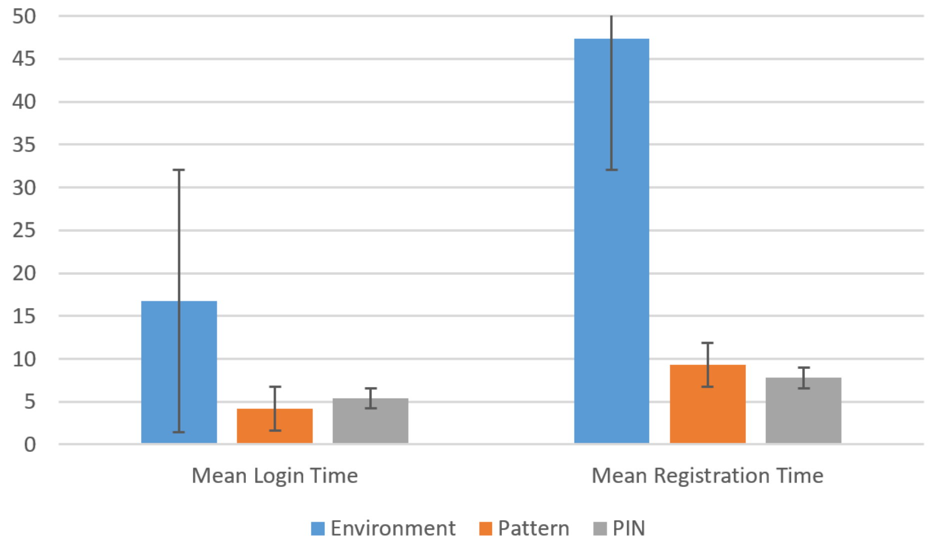


Figure 16: *Mean Login and Registration times in seconds for the three authentication schemes.*

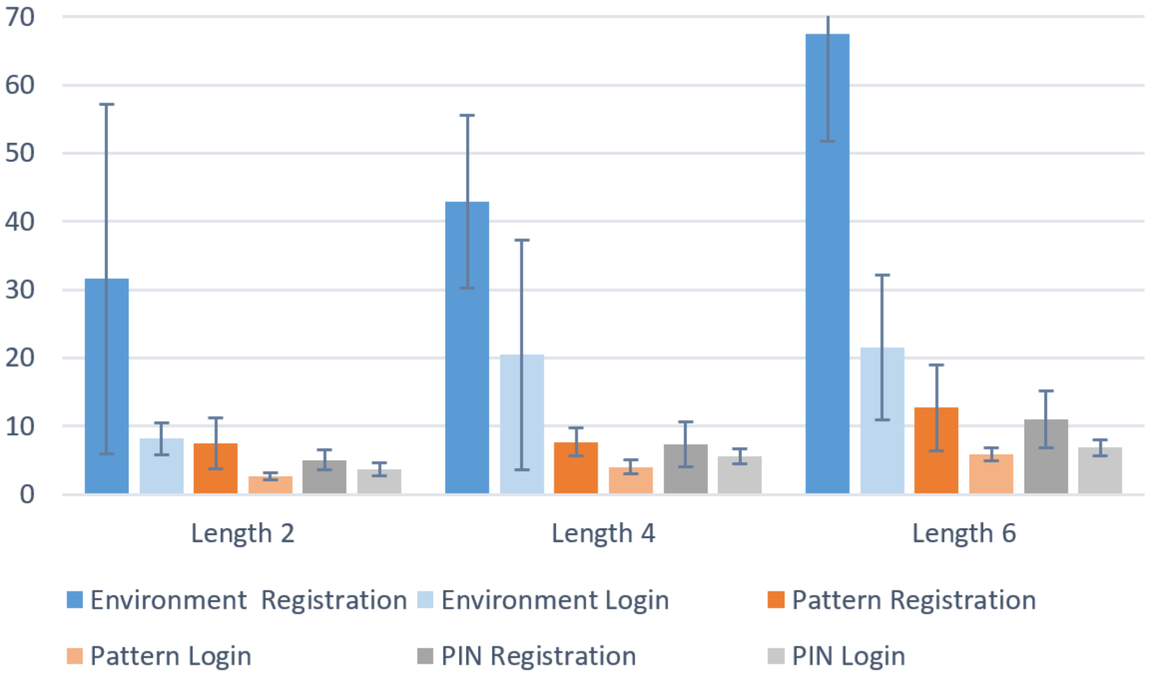


Figure 17: *Mean Login and Registration times in seconds for the three authentication schemes depending on the password length.*

A univariate ANOVA shows that there is a significant interaction between password length, authentication scheme used and experiment step (either *login* or *registration*) with $p < 0.02$ ($F_{(2,198)} = 3.097, \eta^2 = 0.059$). The ANOVA also found a statistically significant interaction of: the password length and used authentication scheme ($p < 0.01, F_{(4,207)} = 4.538, \eta^2 = 0.081$) with the login/registration duration. Password length and experiment step (either login or registration) did not have a significant effect($p < 0.17, F_{(2,210)} = 1.831, \eta^2 = 0.017$). Additionally all three independent variables authentication scheme used ($p < 0.01, F_{(2,213)} = 116.279, \eta^2 = 0.405$), password length ($p < 0.01, F_{(2,213)} = 6.666, \eta^2 = 0.059$) and experiment step (login/registration) ($p < 0.01, F_{(1,214)} = 27.687, \eta^2 = 0.115$) show a statistically significant effect on their own.

6.2 User Errors

There are two different kinds of user errors that are tracked in my implementation of the three authentication schemes. A noticed user-error is recorded whenever a participant reverts her last action in the authentication scheme which can occur in both registration and login processes. An unnoticed mistake can only occur in the login phase and it marks a failed authentication attempt, which means that the user entered a wrong password.

In the following paragraphs an unnoticed error will be referenced as a fail and a noticed mistake will be referenced as an UNDO.

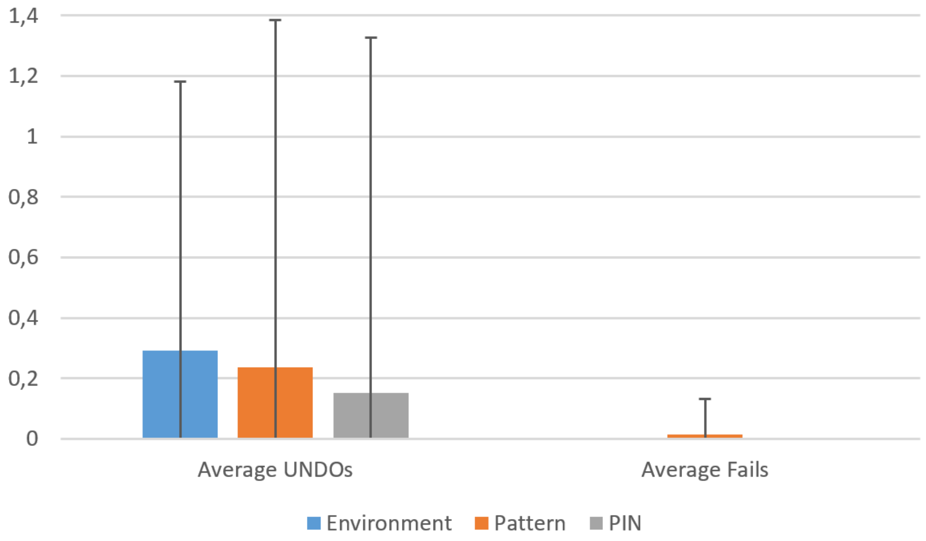


Figure 18: *Mean number of noticed mistakes (UNDOs) and unnoticed mistakes (fails) for the three authentication schemes.*

In figure 18 one can see, that users performed the most UNDOs in the PEA scheme with 0.29 ($SD = 0.89$) on average followed by the Android Pattern scheme with 0.24 ($SD = 1.15$) and the PIN scheme with just 0.15 ($SD = 1.17$). Overall the number of UNDOs that were performed is very low for all three authentication schemes.

A single fail occurred in the Android Pattern authentication, which resulted in a mean of 0.01 ($SD = 0.12$). Both the PIN and PEA scheme did not register any failed login attempts as shown in figure 18.

A univariate ANOVA shows that there is no significant effect on the number of UNDOs for the different authentication schemes ($p < 0.75, F_{(2,213)} = 0.298, \eta^2 = 0.003$) and also no significant effect on the number of fails ($p < 0.37, F_{(2,213)} = 1, \eta^2 = 0.009$).

6.3 Attacking the Authentication Schemes

As shown in figure 19, the attacker success rate varied greatly for the three different authentication schemes when averaged over the password lengths two, four and six. The attacker had a success rate of 5.56% when attacking a password in the PEA scheme followed with a huge gap by the PIN scheme on which the attacker had a success rate of 63.89% and the highest success rate of 94.33% in the android pattern authentication scheme.

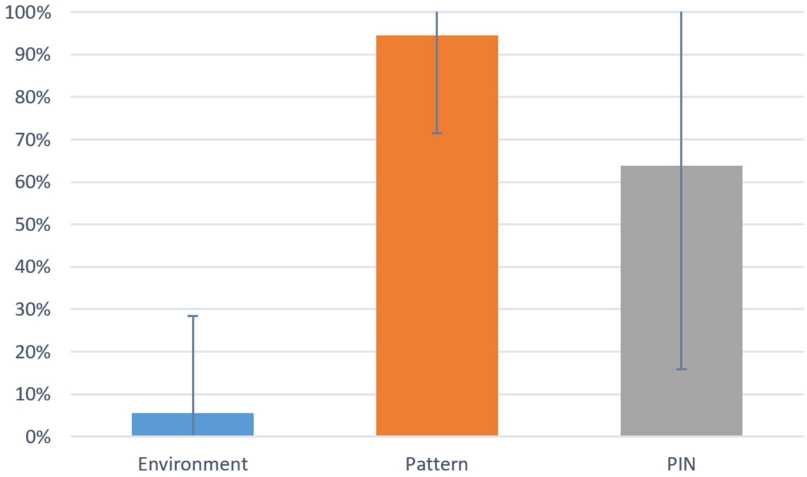


Figure 19: *Success rate of the attacker in the three different authentication schemes.*

The univariate ANOVA detects a statistically significant difference for the authentication schemes ($p < 0.01, F_{(2,213)} = 129.432, \eta^2 = 0.549$) regarding the success rate of the attacker.

Looking at the mean attacker success rate distributed over the password sizes two, four and six as shown in figure 20 you can see that 16.67% ($SD = 0.37$) of the PEA scheme

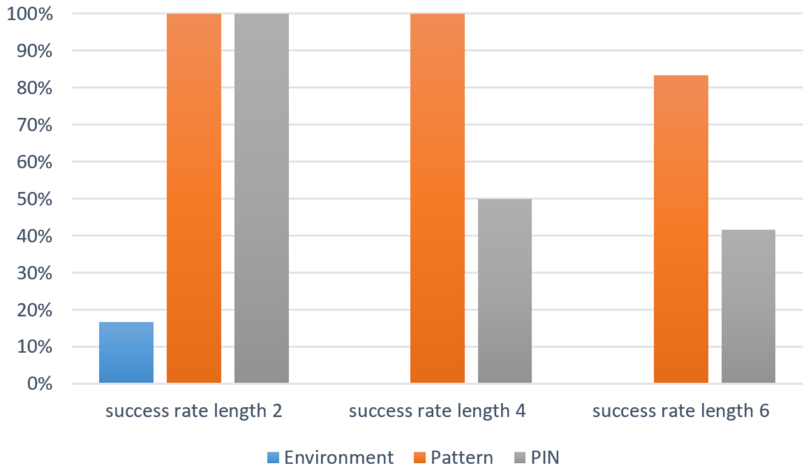


Figure 20: *Success rate of the attacker in the three different authentication schemes for the password sizes 2, 4 and 6.*

passwords of size two have been attacked successfully while the attack failed on all passwords of size four and six in the PEA scheme.

In the android pattern authentication scheme, the attacker was able to successfully attack all passwords of size two and four and 83.33% ($SD = 0.37$) of the passwords of size six.

The passwords of length two in the PIN scheme have all been attacked successfully as well, while the attacker was able to successfully attack 50.0% ($SD = 0.50$) of the passwords of size four and 41.67% ($SD = 0.49$) of the passwords of size six.

A univariate ANOVA shows that there is also a significant effect on the dependent variable 'attacker success rate' for the different password lengths ($p < 0.01, F_{(2,213)} = 7.661, \eta^2 = 0.067$). Additionally a significant interaction between authentication scheme, password length regarding the attacker success rate ($p < 0.01, F_{(4,207)} = 5.802, \eta^2 = 0.101$) was found using an univariate ANOVA.

There are also very large differences between the three authentication schemes and different password lengths regarding the average time an attack took on a particular scheme, as visualized in figure 21. A successful attack on the environment scheme with a password of size two took 57.86 seconds ($SD = 16.14$) on average, which is only surpassed by a successful attack on a PIN password of length 6, which took 75.14 seconds ($SD = 17.30$) on average.

PIN passwords of length two and four and android pattern passwords of length 6 were all successfully attacked with less than 30 seconds on average, which is only a forth of the time the attacker was allowed to use. Android pattern passwords of length six took

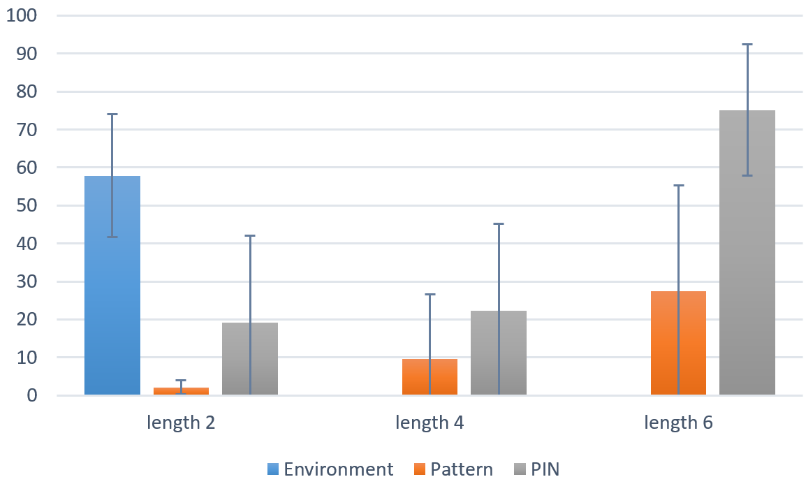


Figure 21: *Average time a successful attack took on the three different schemes depending on length of the password.*

on average 27.37 seconds ($SD = 27.87$) to break followed by successful attacks on PIN passwords of lengths four and two which only took 22.29 seconds ($SD = 22.81$) and 19.17 ($SD = 22.87$) seconds on average.

The passwords which took by far the least amount of time to break on average were android pattern passwords of length four, which already took less than 10 seconds to break on average with 9.65 seconds ($SD = 17.75$) and android pattern passwords of length two which only took 2.14 seconds ($SD = 1.74$) to successfully attack on average.

A univariate ANOVA shows that there is a significant effect on the dependent variable 'attacker-time' for the different authentication schemes ($p < 0.01, F_{(2,213)} = 141.290, \eta^2 = 0.570$), the three different passwords lengths ($p < 0.01, F_{(2,213)} = 13.980, \eta^2 = 0.116$) and also a significant interaction between authentication scheme and password length ($p < 0.01, F_{(4,207)} = 11.976, \eta^2 = 0.188$).

Another interesting observation was, that 25 of the 37 successful attacks on android pattern passwords were broken with the first try of the attacker. That means that the attacker had a 67.57% probability to break a android pattern password on the first try. To put that into perspective only 2 of the 26 successful attacks on the PIN scheme and zero of the two successful attacks on the PEA scheme were performed with the first try the attacker made.

6.4 User Experience

The user experience of the three different schemes was measured after the participant performed all authentications in the three different schemes using the User Experience Questionnaire (UEQ)[14] which contains 27 questions which can be answered on a scale from -3 to $+3$. [38]

These 27 properties can be summarized using the following six properties: attractiveness, perspicuity, efficiency, dependability, stimulation and novelty. [38]

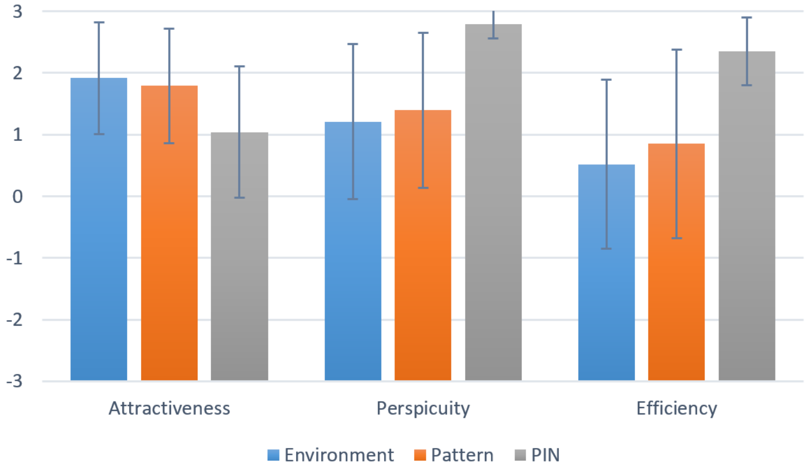


Figure 22: *Attractiveness, perspicuity and efficiency score of the authentication schemes resulting from UEQ.*

As shown in figure 22 the environment scheme was rated the most attractive, which indicates that users overall liked it the most, with an average score of 1.92 ($SD = 0.90$) followed by the android pattern scheme with an average score of 1.79 ($SD = 0.93$) and quite far behind the PIN scheme with an average attractiveness of just 1.04 ($SD = 1.07$).

Regarding perspicuity, which summarizes how easy it is to get familiar with the authentication scheme, the order in which the authentication schemes placed is reversed, with the PIN scheme scoring by far the highest with an average of 2.79 ($SD = 0.23$) followed by the android pattern scheme with 1.40 ($SD = 1.26$) and the PEA scheme with 1.21 ($SD = 1.26$).

The schemes got ranked in the same order regarding their efficiency, which describes how efficient (no unnecessary effort necessary to perform the task) users were able to authenticate. PIN was rated by far the most efficient with an average of 2.35 ($SD = 0.55$)

[14] www.ueq-online.org/

followed by the android pattern scheme with 0.85 ($SD = 1.53$) and the PEA scheme with 0.52 ($SD = 1.37$).

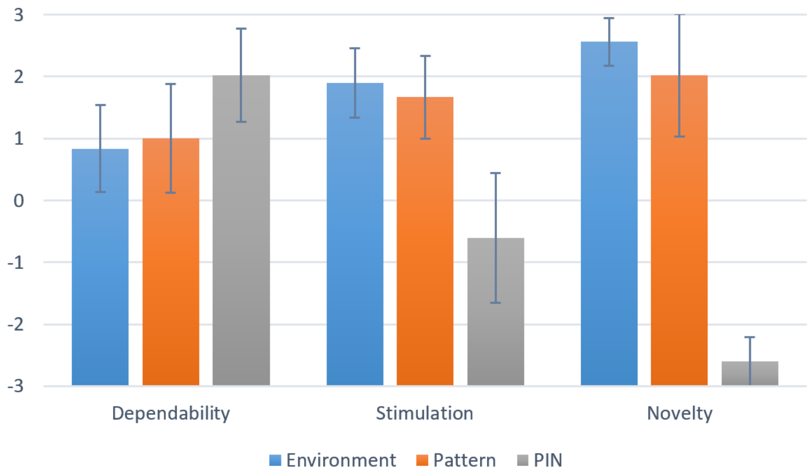


Figure 23: *Dependability, stimulation and novelty score of the authentication schemes resulting from UEQ.*

The same ranking occurred again when rating the scheme according to their dependability as shown in figure 23. A high dependability indicates that the user always feels in control of the interaction with the authentication scheme. PIN had the highest dependability score with an average of 2.02 ($SD = 0.76$) followed by the android pattern scheme with an average score of 1.00 ($SD = 0.88$) and the PEA scheme with an average of 0.83 ($SD = 0.70$).

The PIN scheme was rated by far the lowest stimulation score, with an average of just -0.60 ($SD = 1.05$) which means that it is the least exciting of the authentication schemes, and not motivating to the user. PEA scheme scored the highest with an average stimulation score of 1.90 ($SD = 0.56$) closely followed by the android pattern scheme with a score of 1.67 ($SD = 0.67$).

In the novelty category, the PIN scheme is again ranked by far the worst with an average score of just -2.60 ($SD = 0.39$). The environment scheme had the highest average novelty score with 2.56 ($SD = 0.38$) which means that it is the most innovative and creative. The android pattern scheme also scored very high, with an average of 2.02 ($SD = 0.99$).

A multivariate ANOVA performed on all these six usability properties used as dependent variables showed that the authentication scheme has a statistically significant effect on attractiveness ($p < 0.01, F_{(2,213)} = 17.364, \eta^2 = 0.140$), perspicuity ($p < 0.01, F_{(2,213)} = 47.211, \eta^2 = 0.307$), efficiency ($p < 0.01, F_{(2,213)} = 43.160, \eta^2 = 0.288$), dependability

($p < 0.01, F_{(2,213)} = 56.165, \eta^2 = 0.345$), stimulation ($p < 0.01, F_{(2,213)} = 222.751, \eta^2 = 0.677$) and novelty ($p < 0.01, F_{(2,213)} = 1283.065, \eta^2 = 0.923$).

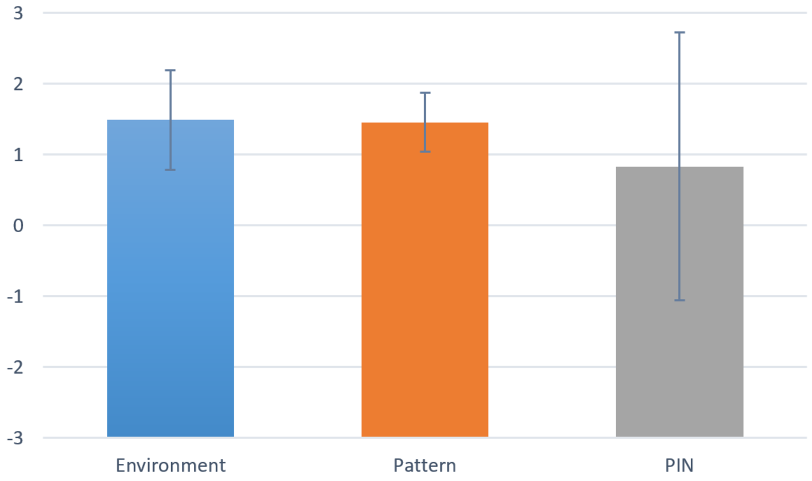


Figure 24: *Overall user experience score of the three authentication schemes.*

The highest average score in all six categories is achieved by the environment scheme, with an overall user experience score of 1.49 ($SD = 0.70$) closely followed by the android pattern scheme with a score of 1.45 ($SD = 0.42$). The PIN scheme performed by far the worst regarding user experience with an average score of 0.83 ($SD = 1.89$) as shown in figure 24. The overall user experience score also has a significant effect as confirmed by a univariate ANOVA ($p < 0.01, F_{(2,213)} = 23.225, \eta^2 = 0.179$).

6.5 Workload

To measure the workload the participants experienced in the different authentication schemes during the study, they answered a short non-weighted NASA TLX questionnaire developed by Jens Grubert[15] (based on the original NASA TLX[16]) separately for each of the authentication schemes.

The PIN scheme scored the best (lower score) in all five categories, while the android pattern authentication scheme was the second best in every category, except frustration, in which the environment scheme scored slightly better. Overall the PIN scheme achieved the lowest (best) workload with an average score of just 3.78 ($SD = 1.92$) followed by the android pattern scheme with an average workload of 5.44 ($SD = 2.91$) and the environment scheme with an average workload of 8.32 ($SD = 3.64$) as seen in figure 25.

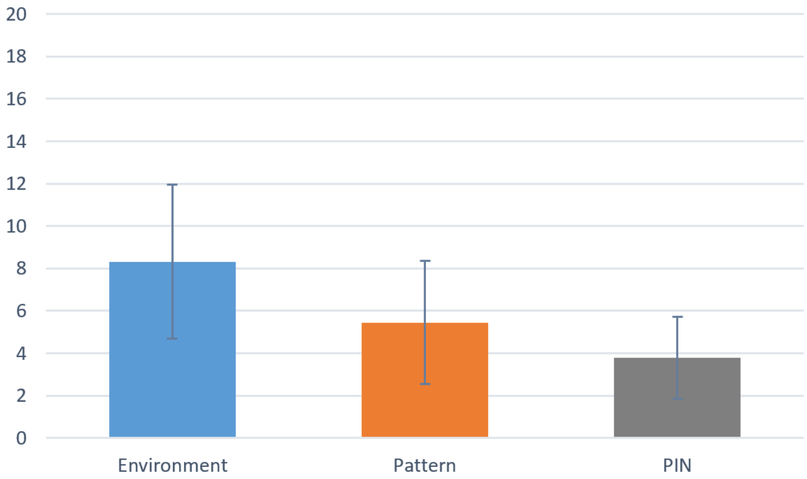


Figure 25: *Overall workload score of the three authentication schemes.*

A univariate ANOVA shows that there is a statistically significant effect on the average workload for the different authentication schemes with $p < 0.01$ ($F_{(2,213)} = 44.320, \eta^2 = 0.294$).

[15] https://jensgrubert.wordpress.com/2014/09/01/nasa-tlx-short-non-weighted-version-in-html-javascript/

[16] humansystems.arc.nasa.gov/groups/TLX/

6.6 Demographics

The demographics questionnaire I developed asked people to rate their experience and how comfortable they are with the three authentication schemes from one (no experience) to seven (very experienced). Users were the most experienced in the PIN scheme with an average score of 7.0 ($SD = 0$) closely followed by the Android Pattern scheme with an average of 6.67 ($SD = 0.62$). The PEA had by far the lowest experience with an average of just 1.17 ($SD = 0.37$). Users were the most comfortable in the PIN scheme with an average score of 6.83 ($SD = 0.37$) closely followed by the Android Pattern scheme with a score of 6.00 ($SD = 1.87$) and PEA scheme with a score of 5.67 ($SD = 1.55$) as shown in figure 26.

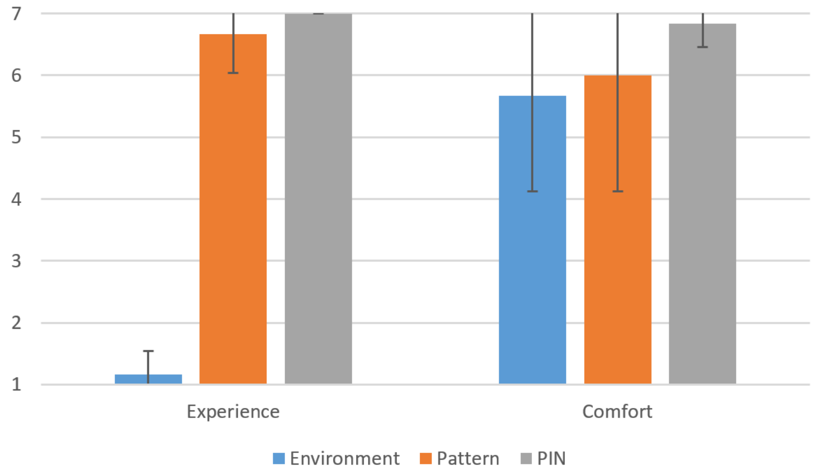


Figure 26: *Average experience and comfort with the authentication schemes.*

Two univariate ANOVAs performed showed that the authentication scheme has a statistically significant effect on the previous experience ($p < 0.01, F_{(2,213)} = 4331.000, \eta^2 = 0.976$) and the comfort ($p < 0.01, F_{(2,213)} = 12.760, \eta^2 = 0.107$) with the scheme.

In the same questionnaire the participants were also asked how secure they expect the schemes to be from one (very insecure) to seven (perfect security) and how easy it is to memorize a password created with the schemes from one (very hard to remember) to seven (very easy to remember). The PEA scheme scored the highest average security score with 5.92 ($SD = 0.76$) followed by the PIN scheme with 4.75 ($SD = 1.79$ and the Android Pattern Scheme with an average score of 4.08 ($SD = 0.76$). My newly developed scheme also scored the highest for the memorability of its passwords with an average score of 6.00 ($SD = 0.82$) followed by the Android Pattern scheme with a score of 4.58 ($SD = 1.71$) and the PIN scheme with 3.42 ($SD = 1.66$) as seen in figure 27.

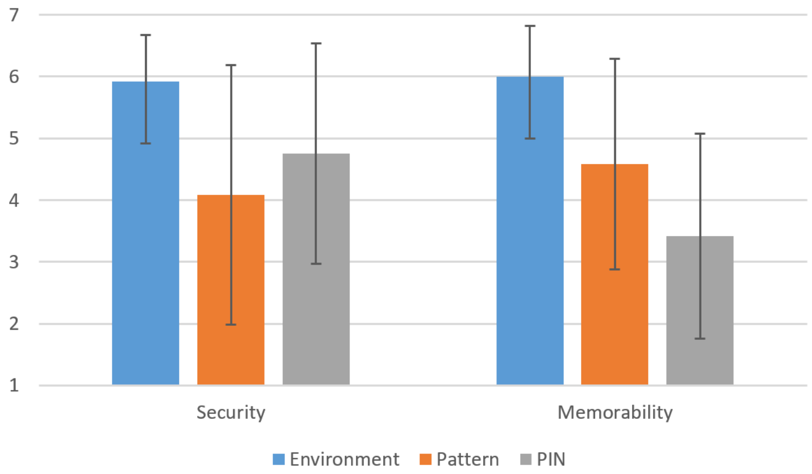


Figure 27: *Average participant-rated security and memorability of the authentication schemes.*

Another two univariate ANOVAs performed showed that the authentication scheme has a significant effect on the participant-rated security ($p < 0.01, F_{(2,213)} = 22.440, \eta^2 = 0.174$) and memorability ($p < 0.01, F_{(2,213)} = 56.410, \eta^2 = 0.346$) of the authentication schemes.

7 Discussion

7.1 Registration and Login Time

Looking at the registration and login times of the three different authentication schemes, it is very obvious that my newly developed authentication scheme had the worst mean task completion times. According to the concept of Fitts law, this was to be expected. [39] Especially the very long time it took to register in the PEA scheme stands out compared to the PIN and Android Pattern authentication scheme. The ANOVA confirmed that the PEA scheme is the slowest to authenticate with, followed by the PIN scheme and lastly the Android Pattern scheme.

When looking at figure 16 it is noticeable that the registration in the PEA scheme is very slow even compared to its already slow login time. The registration takes almost three times as long as the login, while it only takes around 2 times as long for the Android Pattern scheme and around 1.5 times as long for the PIN scheme. An explanation for the large difference between registration and login times in my newly developed authentication scheme are the additional steps a user has to perform in the registration, e.g. scaling spheres to the desired size using swipe gestures and the potential creation of custom shapes. It is quite obvious that there are a lot more interactions necessary to create a password of length six in the PEA scheme, compared to just authenticating with the same password. Even if the user does not use any custom shapes and all spheres are left at their default size, the registration would still take significantly longer because every sphere has to be confirmed after positioning it using either a swipe to the right or a press on the confirm button. Additionally the user has to press the save password button at the very end which is not necessary in the login phase.

Another factor which explains a longer registration time is the fact that precision is more important in the registration phase to achieve a good coverage of the password objects using the spherical or custom shapes. In the login phase, it is much more tolerant to small imprecisions because you can click anywhere in the previously created sphere which usually is a fairly large area and does not require a lot of precision to hit.

The overall significantly higher mean time it takes to register and login in the PEA scheme, compared to other methods, is not very surprising as it requires a lot more movement by design. Many users, especially those with limited experience in VR, turned slowly which increased the time it took to authenticate. It might be possible that the sitting posture slowed the turning down compared to a standing posture. The fact that randomly generated passwords were used in the study increased this effect even further, because randomly chosen objects were placed far from each other on average which introduces a lot of movement in between selecting the objects. In the real world a user who prefers a relatively quick login time would be able to create a password with objects close to each other to avoid this problem. Another slowdown caused by the randomly generated passwords in my scheme was that users regularly searched for objects in the picture which were part of the randomly generated password.

I also assume that lack of experience played an important role in slowing down users in the PEA scheme. Most people would probably authenticate a lot more efficiently and quickly after more practice with the authentication scheme.

Figure 17 indicates that comparison of the mean times of the PEA scheme with a password length of two to the mean times of the PIN and Android Pattern schemes with the same password length, might not be reasonable. The attacks on the three authentication schemes showed that even a PEA password of size two is a lot more secure than passwords of length six in both the PIN and Android Pattern authentication scheme.

7.2 User Errors

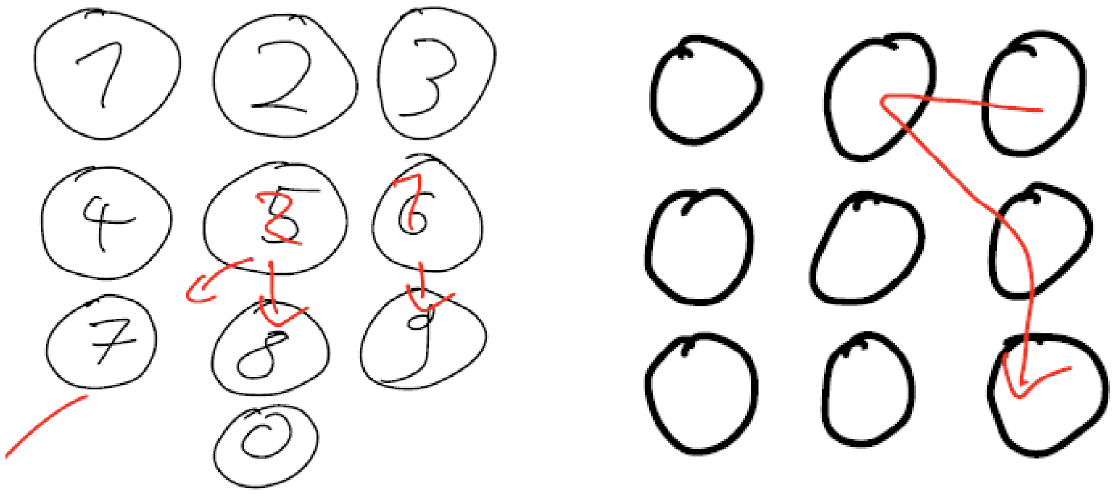
Overall, the number of user errors that occurred was very low for all three authentication schemes, and the difference between the schemes was proven to be insignificant for both noticed and unnoticed user-errors.

These results prove that I was successful in providing good and helpful feedback to the users of my novel authentication scheme. I also expect the number of UNDOs to decrease significantly for users with more experience in the authentication scheme than just the short tutorial session of my study. Clearly it is a good sign that even very inexperienced users were able to use the scheme without many user errors.

Additionally it proved that even the suboptimal precision offered by the Google Daydream Controller is absolutely sufficient to authenticate using the PEA scheme.

7.3 Attacking the Authentication Schemes

The results discovered in my study clearly show that the PEA scheme is by far the most secure against shoulder surfing attacks which was confirmed by a univariate ANOVA. It is quite remarkable how insecure the PIN and Android Pattern authentication schemes are against these attacks, especially considering their popularity. The attacker developed advanced strategies to attack all three authentication schemes and I conducted an interview with him before and after attacking all the passwords participants entered during the study.



(a) *PIN Scheme*

(b) *Android Pattern Scheme*

Figure 28: *Example notes of attacker.*

For both the PIN scheme and the Android Pattern scheme a similar strategy was used as shown in figure 28. Essentially, the attacker drew either the numpad for the PIN scheme or a three times three grid for the Android Pattern scheme on a sheet of paper in the preparation phase. He used these templates to draw the movement the study participant was making while watching the video of the login. While it was generally very easy for the Android Pattern scheme, the PIN scheme was significantly harder because it was difficult to differentiate a movement to the first or second row. For the Android Pattern scheme this problem was nearly irrelevant because of the continuous movement that has to represent the password by design. Whenever the attacker was not certain, which number was pressed, he made additional notes indicating which alternatives seemed possible given the observed movement.

Whenever the password did not cover all rows or columns of the grid or numpad, the attacker used a sliding window approach to brute-force all possible passwords in a very short amount of time.

For the PEA scheme, the attacker had a much harder time to develop an advanced strategy to attack the passwords. The final strategy, after extensive testing, was to estimate the degrees of rotation for both the x and y axis, that were performed by a participant between each click on the picture. The attacker then tried to replicate the rotations in a sliding-window inspired approach as a brute-force attack. He mentioned that attacking passwords, created with this scheme, felt very much like guessing and that he generally had no idea if he was anywhere near to the correct password. The primary reason was the difficulty to estimate the degrees of rotation that were performed with the controller, because there are no boundaries in the 360-degree picture used in the PEA scheme. He felt that breaking this scheme is only feasible for passwords of length two if the used objects are both reasonably large and very close to each other (or even the same object) because this allowed for very fast brute force attacks using a sliding-window approach. The attacker concluded that the PEA scheme is already extremely secure against shoulder surfing attacks with a password size of four, even when the attacker is given unlimited time to attack and estimates the performed rotations reasonably precise because of the pure number of possible combinations. Both the PIN and Pattern schemes are very easy to attack for all the tested lengths and with more time than just five minutes, he feels confident that he can break all passwords using these approaches.

Given a scenario in which the attacker has a very strict time limitation or a strictly limited amount of tries, my testing also showed that the Android Pattern scheme performs by far the worst as 25 of 37 successful attacks were performed on the first try of the attacker. In general the time it took to break passwords created with the Android Pattern scheme is extremely low even compared to the PIN scheme.

Overall the security of the Android Pattern and PIN scheme against an experienced shoulder surfing attacker is very disappointing up to the tested password length of six and not usable whenever a user would like to protect private data etc.

7.4 User Experience

Overall the PEA scheme scored the highest user experience score, closely followed by the Android Pattern scheme, while the PIN scheme scored the worst in terms of overall user experience. As all three scores are significantly above zero, the user experience can generally be considered good for all three authentication schemes. Notable differences in the different user-experience categories are the significantly higher score in perspicuity, efficiency and dependability of the PIN scheme compared to both the Android Pattern and PEA scheme. This is not very surprising considering that every single participant was already familiar and very comfortable with the PIN scheme before taking part in my experiment. This led to the users being in full control all the time and solving all the tasks very efficiently. These three high scores were followed by a low stimulation score and an extremely low novelty score, which are again not very surprising as all the participants have authenticated using this scheme countless times already.

The Android Pattern and PEA schemes are not only very close in their overall user-experience score, but also in all six categories. The only significant difference was that participants found the PEA scheme to be significantly more innovative and creative than the Android Pattern scheme.

In a small voluntary discussion I had with participants after the experiment regarding disadvantages or problems of my newly developed authentication scheme many users mentioned that the Daydream Controller was not very precise, which led to some frustration. Additionally the low resolution of the HMD made authentication in the PEA scheme much less pleasant. Many participants also mentioned that more experience with using the scheme would probably improve its score in terms of usability much further and that it is generally better suited for usage in VR compared to the 'Android Pattern' and 'PIN' scheme. They felt more motivated and attracted by the approach of the PEA scheme.

7.5 Workload

The overall workload was significantly different for all three authentication schemes. My newly developed scheme scored significantly worse than both the Android Pattern and PIN scheme, possibly due to its VR-suited design. It is very obvious that authentication in the PEA scheme requires much more movement and also significantly more time, which is why it scored significantly higher physical and temporal demand. The significantly higher effort is probably also caused by the additional actions one has to perform to either register or authenticate using my newly developed scheme compared to both the Android Pattern and the PIN scheme. I explained all these additional efforts in section 7.1.

7.6 Demographics

Although all thirteen participants had no experience with the PEA scheme, users felt almost as comfortable in it as in the Android Pattern scheme. Overall my scheme seemed to not be overly complex and did not overtax the user.

Even after this short amount of time, participants were convinced that the PEA scheme is significantly more secure than both alternatives. The reason for that is most likely the number of possible passwords with a detailed background panorama picture.

Additionally the participants also rated the passwords of my scheme to be the easiest to memorize. They potentially felt that graphical passwords in general are much easier to memorize than a numeric code or a gesture as used in the Android Pattern scheme. The participants expected to have the hardest time remembering simple numeric codes as used in the PIN scheme. Overall these results look exactly as I expected them from the related work I studied, which is described in section 2.5 of this thesis.

8 Conclusion

The PEA scheme is an effort to develop security innovations specifically for usage in VR. It improves upon existing solutions mainly by being a graphical authentication scheme, which makes use of the increased immersion VR can offer compared to alternative media. The study proved that the PEA scheme was significantly more resistant against shoulder surfing attacks from an expert attacker than both the PIN and Android Pattern authentication scheme. The participants of my study also rated it a as a more pleasant user experience than both established alternatives despite them being entirely inexperienced to the approach of the PEA scheme, as opposed to extensive experience with both authentication schemes it was compared to.

Additionally the study discovered that authentication in my newly developed authentication scheme is significantly slower and requires much more (physical) work. This disadvantage can be weakened by using a much shorter password as opposed to alternative studies. This can be done, because even passwords of length two in the Personal Environment Scheme were found to be significantly more secure than passwords of length six in both the PIN and Android Pattern scheme. For sufficient security against an experienced shoulder surfing attacker, the passwords would need to be much longer than the maximum length I tested in my experiment for both the PIN and Android Pattern scheme. Meanwhile a password length of four in the PEA scheme seems to be sufficiently secure against an expert attacker, assuming the user uses a personal picture with a reasonable amount of details.

The biggest advantage of the PEA, compared to all established methods of authentication, is most likely the longterm memorability of passwords created using personal images as environment. Unfortunately it was not possible to verify this in an experiment due to time constraints of a bachelor thesis, but research regarding the Pictorial Superiority Effect [13, 16, 28] and the answers of study participants in questionnaires support this assumption.

9 Future Work

Extensive long-term Memorability Analysis

To verify the expected long-term memorability my newly developed scheme provides, one would have to perform a large study over a much longer timespan. For my thesis this was unfortunately not possible due to obvious time constraints.

Expansion to different VR Hardware

Many users mentioned either the fairly imprecise Google Daydream Controller or the low resolution of the Daydream HMD as a disadvantage of the PEA scheme. In order to conform or deny this assumption one could port the authentication scheme to more advanced hardware like the HTC Vive or Oculus Rift and conduct additional studies on this hardware.

Translation to AR Scheme

The PEA scheme can be transfered to the field of AR with a few minor changes. Instead of displaying a full skybox consisting of a 360 degree equirectangular image one would just display a variety of objects in AR which have to be selected in the correct order. In an ideal case they would seamlessly fit into the real environment (lighting conditions, shadows etc.).

Obviously it would be much more work to create these objects with the technology available today. One would have to model these objects separately instead of simply taking a equirectangular image and importing it to my authentication scheme.

While the performance in AR would have to be tested, I expect it to be significantly worse in terms of password memorability compared to its VR counterpart, mainly because the user is not able to use a completely personal image of e.g. his living room. Limiting the scheme to multiple objects will most likely make it harder to associate the set of objects with the story behind the password, because there is much less information contained in these objects compared to an entire picture of the user's choice.

Replacing Static Image with Interactive Environment

An extension to the approach, explained in the previous paragraph, would be to model the entire surrounding as three-dimensional objects. This would add the possibility to interact with certain objects during the authentication which has proven to be feasible by Alsulaiman et al [40, 41]. An example would be opening a drawer and clicking on some object inside the drawer as part of a password. The additional options, this would introduce to the authentication scheme, are substantial. Considering recent progress in computer vision, it might be possible in the future to automatically generate these three-dimensional objects from pictures instead of modeling every object by hand.

An interactive environment would not only increase the number of possible passwords significantly (and thus increase the security of the scheme), it would also increase the immersion of the authentication scheme, as the user would be able to interact with objects in the virtual world just like she could in the real world. [1]

Security Improvement using Multi-Factor Authentication

Just like most authentication schemes, it would be possible to implement an additional authentication step using Multi-Factor Authentication in the PEA scheme.

Although this would require the user to use additional hardware and likely to take off his HMD, it could be used to protect security critical information. One could use a simple one-time numeric password, generated on e.g. the user's smartphone [42], or much more sophisticated approaches. In the publication "Universal Multi-Factor Authentication Using Graphical Passwords" by Sabzevar et al. hint messages were transmitted to the user's handheld device to communicate information to the user which was necessary to authenticate with a graphical authentication scheme. [30]

Additionally an extension could be implemented which allows the user to add a story to every password. An attacker would have to know the correct coordinates to click with their specific order as well as the story. This would allow additional personalization of the password only understood by the user himself.

Example: The user uses a picture of his living room as the environment for the PEA scheme. His password is a sequence of seven objects distributed over the entire room. The story behind this password is the selection of gifts he got from friends on vacation in reversed chronological order. Now possible modifications to this transmitted to the handheld device of the user could be: "no Peter", which would mean to leave out Peter's present in the password sequence; or "reversed starting at grandmother" which would mean to revert the password sequence starting at the present the user got from his grandmother. It is quite clear that there is a large number of modifications that can be communicated to the user's handheld device, which leak no relevant information about the password to an attacker intercepting this message.

Acknowledgement

I would like to thank Marco Speicher for his work as my supervisor, assisting me in my first user-study and the writing process of my thesis. I would also like to thank the entire team of the Ubiquitous Media Technology Lab, who were always keen to help when I had any issues.

I would also like to thank Prof. Dr. Antonio Krüger for giving me the opportunity to write my bachelor thesis at the Research Center for Artificial Intelligence in Saarbrücken, Germany.

Special thanks also go to my girlfriend and family, especially my brother for always assisting me as an application tester and for their moral support throughout the process of writing my thesis.

Last but not least, I would like to thank every participant of my study again.

List of Figures

1 KinWrite successfully verified the genuine signature and rejected the forged handwriting. [11] . . . 14
2 Prototype of the multi-sensor based framework called Pingu. [12] . . . 14
3 Password sequence 3-left-4-right. [25] . . . 17
4 Success rate on attacking GazeTouchPass which shows the decreasing success rate with increasing the number of modality switches. [25] . . . 18
5 Example selection of both authentication schemes. [18] . . . 21
6 Example password of the PassPoints scheme. [19] . . . 22
7 User Interface of the PEA scheme. [32] . . . 27
8 Picture of Google Daydream View. [33] . . . 29
9 Picture of the Google Pixel (Gen 1). [34] . . . 30
10 Interface of the Google Daydream Controller. [35] . . . 30
11 Rescaling before and after. [32] . . . 32
12 Addition of Swipe Menu to Daydream Controller in VR. . . . 33
13 The problems of spherical shapes. [32] . . . 35
14 User created custom shape using four clicks to cover the drawing. [32] . . 36
15 Screenshots of my VR implementation of PIN and Android Pattern scheme. 43
16 Mean Login and Registration times in seconds for the three authentication schemes. . . . 49
17 Mean Login and Registration times in seconds for the three authentication schemes depending on the password length. . . . 49
18 Mean number of noticed mistakes (UNDOs) and unnoticed mistakes (fails) for the three authentication schemes. . . . 50
19 Success rate of the attacker in the three different authentication schemes. 51
20 Success rate of the attacker in the three different authentication schemes for the password sizes 2, 4 and 6. . . . 52
21 Average time a successful attack took on the three different schemes depending on length of the password. . . . 53
22 Attractiveness, perspicuity and efficiency score of the authentication schemes resulting from UEQ. . . . 54
23 Dependability, stimulation and novelty score of the authentication schemes resulting from UEQ. . . . 55
24 Overall user experience score of the three authentication schemes. . . . 56
25 Overall workload score of the three authentication schemes. . . . 57
26 Average experience and comfort with the authentication schemes. . . . 58
27 Average participant-rated security and memorability of the authentication schemes. . . . 59
28 Example notes of attacker. . . . 62
29 English User Experience Questionnaire. . . . 78

30 NASA TLX Questionnaire. 79
31 First page of the demographics questionnaire. 81
32 Second page of the demographics questionnaire. 82
33 Third page of the demographics questionnaire. 83
34 Forth page of the demographics questionnaire. 84
35 Fifth page of the demographics questionnaire. 85
36 Sixth page of the demographics questionnaire. 86
37 Seventh page of the demographics questionnaire. 87
38 Eighth page of the demographics questionnaire. 88
39 Ninth page of the demographics questionnaire. 89

Abbreviations

VR

Virtual Reality

AR

Augmented Reality

HMD

Head Mounted Display

PIN

Personal Identification Number

PEA

Personal Environment Authentication

UI

User Interface

3D

Three-Dimensional

2D

Two-Dimensional

UEQ

User Experience Questionnaire

NASA-TLX

NASA Task Load Index

Bibliography

[1] Ralf Dörner, Bernhard Jung, Paul Grimm, Wolfgang Broll, and Martin Göbel. *Einleitung*, pages 1–31. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[2] Lawrence J. Hettinger and Gary E. Riccio. Visually induced motion sickness in virtual environments. *Presence: Teleoperators and Virtual Environments*, 1(3):306–310, 1992.

[3] James P. Bliss, Philip D. Tidwell, and Michael A. Guest. The effectiveness of virtual reality for administering spatial navigation training to firefighters. *Presence: Teleoperators and Virtual Environments*, 6(1):73–86, 1997.

[4] Kelvin Valentino, Kevin Christian, and Endra Joelianto. Virtual reality flight simulator.

[5] Ralf Dörner and Frank Steinicke. *Wahrnehmungsaspekte von VR*, pages 33–63. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[6] Denis Besnard and Budi Arief. Computer security impaired by legitimate users. *Comput. Secur.*, 23(3):253–264, May 2004.

[7] Robert Morris and Ken Thompson. Password security: A case history. *Commun. ACM*, 22(11):594–597, November 1979.

[8] Arvind Narayanan and Vitaly Shmatikov. Fast dictionary attacks on passwords using time-space tradeoff. In *Proceedings of the 12th ACM Conference on Computer and Communications Security*, CCS '05, pages 364–372, New York, NY, USA, 2005. ACM.

[9] James Wayman, Anil Jain, Davide Maltoni, and Dario Maio. *An Introduction to Biometric Authentication Systems*, pages 1–20. Springer London, London, 2005.

[10] Ziming Zhao, Gail-Joon Ahn, Jeong-Jin Seo, and Hongxin Hu. On the security of picture gesture authentication. In *Proceedings of the 22Nd USENIX Conference on Security*, SEC'13, pages 383–398, Berkeley, CA, USA, 2013. USENIX Association.

[11] Jing Tian, Chengzhang Qu, Wenyuan Xu, and Song Wang. Kinwrite: Handwriting-based authentication using kinect.

[12] Mehran Roshandel, Aarti Munjal, Peyman Moghadam, Shahin Tajik, and Hamed Ketabdar. *Multi-sensor Finger Ring for Authentication Based on 3D Signatures*, pages 131–138. Springer International Publishing, Cham, 2014.

[13] Joan Gay Snodgrass and Anthony Asiaghi. The pictorial superiority effect in recognition memory. *Bulletin of the Psychonomic Society*, 10(1):1–4, Jul 1977.

[14] E. E. K. Ugochukwu and Y. Y. Jusoh. A Review on the Graphical User Authentication Algorithm: Recognition-based and Recall-based. *Journal of Information Processing and Management*, 4(3):238–252, May 2013.

[15] James M. Clark and Allan Paivio. Dual coding theory and education. *Educational Psychology Review*, 3(3):149–210, Sep 1991.

[16] Robert Biddle, Sonia Chiasson, and P.C. Van Oorschot. Graphical passwords: Learning from the first twelve years. *ACM Comput. Surv.*, 44(4):19:1–19:41, September 2012.

[17] Robert Biddle, Sonia Chiasson, and P.C. Van Oorschot. Graphical passwords: Learning from the first twelve years. *ACM Comput. Surv.*, 44(4):19:3, September 2012.

[18] Darren Davis, Fabian Monrose, and Michael K. Reiter. On user choice in graphical password schemes. In *Proceedings of the 13th Conference on USENIX Security Symposium - Volume 13*, SSYM'04, pages 11–11, Berkeley, CA, USA, 2004. USENIX Association.

[19] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. Passpoints: Design and longitudinal evaluation of a graphical password system. *International Journal of Human-Computer Studies*, 63(1):102 – 127, 2005. HCI research in privacy and security.

[20] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. Authentication using graphical passwords: Effects of tolerance and image choice. In *Proceedings of the 2005 Symposium on Usable Privacy and Security*, SOUPS '05, pages 1–12, New York, NY, USA, 2005. ACM.

[21] F. Aloul, S. Zahidi, and W. El-Hajj. Two factor authentication using mobile phones. In *2009 IEEE/ACS International Conference on Computer Systems and Applications*, pages 641–644, May 2009.

[22] L. O'Gorman. Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*, 91(12):2021–2040, Dec 2003.

[23] A. De Luca and J. Lindqvist. Is secure and usable smartphone authentication asking too much? *Computer*, 48(5):64–68, May 2015.

[24] Shrikala M. Deshmukh and P.R. Devale. An efficient mechanism for secure authentication. *IJCSEITR*, 3(5):85–94, May 2013.

[25] Mohamed Khamis, Florian Alt, Mariam Hassib, Emanuel von Zezschwitz, Regina Hasholzner, and Andreas Bulling. Gazetouchpass: Multimodal authentication using gaze and touch on mobile devices. In *Proceedings of the 2016 CHI Conference*

Extended Abstracts on Human Factors in Computing Systems, pages 2156–2164. ACM, 2016.

[26] Simson Garfinkel Lorrie Faith Cranor. *Security and Usability: Designing Secure Systems that People Can Use*. O'Reilly Media, 1 edition, 2005.

[27] Simson Garfinkel Lorrie Faith Cranor. *Security and Usability: Designing Secure Systems that People Can Use*. O'Reilly Media, 1 edition, 2005.

[28] Antonella De Angeli, Lynne Coventry, Graham Johnson, and Karen Renaud. Is a picture really worth a thousand words? exploring the feasibility of graphical authentication systems. *International Journal of Human-Computer Studies*, 63(1):128 – 152, 2005. HCI research in privacy and security.

[29] Hsin-Yi Chiang and Sonia Chiasson. Improving user authentication on mobile devices: A touchscreen graphical password. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '13, pages 251–260, New York, NY, USA, 2013. ACM.

[30] A. P. Sabzevar and A. Stavrou. Universal multi-factor authentication using graphical passwords. In *2008 IEEE International Conference on Signal Image Technology and Internet Based Systems*, pages 625–632, Nov 2008.

[31] Ceenu George, Mohamed Khamis, Emanuel von Zezschwitz, Marinus Burger, Henri Schmidt, Florian Alt, and Heinrich Hussmann. Seamless and secure vr: Adapting and evaluating established authentication systems for virtual reality, 2017.

[32] Aldo Hoeben. Theo kemp's atelier. `https://flic.kr/p/4Pd447`, 2008. Accessed: 18.03.2018, Creative Commons License, used as equirectangular background picture for Personal Environment Authentication scheme.

[33] WIKIMEDIA COMMONS PUBLIC DOMAIN. Google daydream view (vr). `commons.wikimedia.org/wiki/File:Google_Daydream_View_(VR).jpg`. Accessed: 18.03.2018.

[34] Official Google Store. Pixel, phone by google, 1. generation. `store.google.com/product/pixel_phone`. Accessed: 18.03.2018.

[35] Official Google Support. Daydream view-controller und -headset verwenden. `support.google.com/daydream/answer/7184597`. Accessed: 18.03.2018.

[36] H. Lee, Y. Tateyama, and T. Ogi. Image-based stereo background modeling for cave system. In *2011 IEEE International Symposium on VR Innovation*, pages 251–254, March 2011.

[37] James V. Bradley. Complete counterbalancing of immediate sequential effects in a latin square design. *Journal of the American Statistical Association*, 53(282):525–528, 1958.

[38] Bettina Laugwitz, Theo Held, and Martin Schrepp. Construction and evaluation of a user experience questionnaire. In Andreas Holzinger, editor, *HCI and Usability for Education and Work*, pages 63–76, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[39] Fitts and Paul M. The information capacity of the human motor system in controlling the amplitude of movement. In *Journal of Experimental Psychology: General*, volume 121, pages 262–269, September 1992.

[40] F. A. Alsulaiman and A. El Saddik. Three-dimensional password for more secure authentication. *IEEE Transactions on Instrumentation and Measurement*, 57(9):1929–1938, Sept 2008.

[41] F. A. Alsulaiman and A. E. Saddik. A novel 3d graphical password schema. In *2006 IEEE Symposium on Virtual Environments, Human-Computer Interfaces and Measurement Systems*, pages 125–128, July 2006.

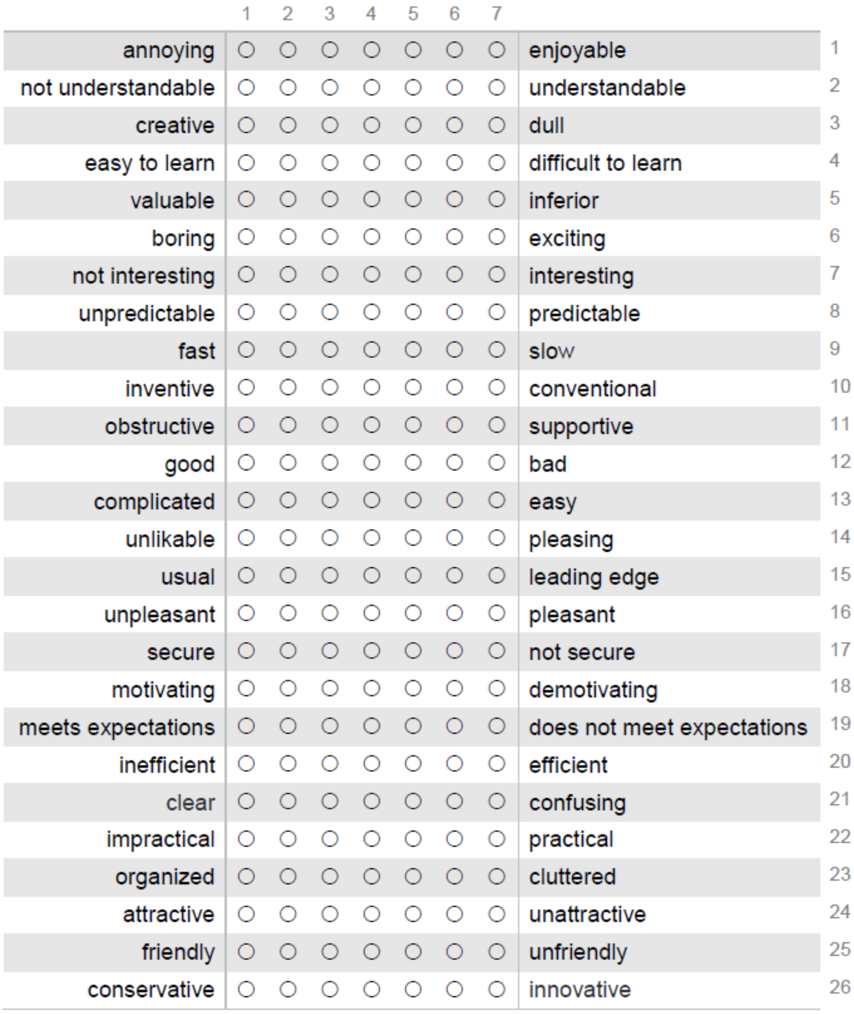
[42] F. Aloul, S. Zahidi, and W. El-Hajj. Two factor authentication using mobile phones. In *2009 IEEE/ACS International Conference on Computer Systems and Applications*, pages 641–644, May 2009.

[43] Sandra G Hart and Lowell E Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. *Human mental workload*, 1(3):139–183, 1988.

Appendices

User Experience Questionnaire (UEQ)

I used the original User Experience Questionnaire (UEQ) by Laugwitz, Held and Schrepp [38] in both German and English. Below you will find the English version attached in figure 29. The same questionnaire was conducted separately for all three authentication schemes.

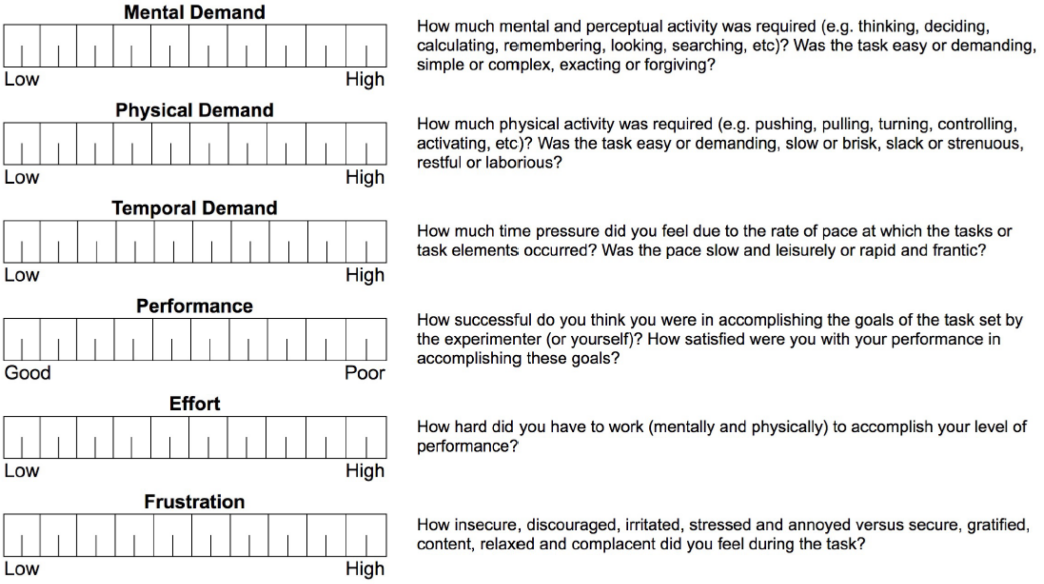


	1	2	3	4	5	6	7		
annoying	○	○	○	○	○	○	○	enjoyable	1
not understandable	○	○	○	○	○	○	○	understandable	2
creative	○	○	○	○	○	○	○	dull	3
easy to learn	○	○	○	○	○	○	○	difficult to learn	4
valuable	○	○	○	○	○	○	○	inferior	5
boring	○	○	○	○	○	○	○	exciting	6
not interesting	○	○	○	○	○	○	○	interesting	7
unpredictable	○	○	○	○	○	○	○	predictable	8
fast	○	○	○	○	○	○	○	slow	9
inventive	○	○	○	○	○	○	○	conventional	10
obstructive	○	○	○	○	○	○	○	supportive	11
good	○	○	○	○	○	○	○	bad	12
complicated	○	○	○	○	○	○	○	easy	13
unlikable	○	○	○	○	○	○	○	pleasing	14
usual	○	○	○	○	○	○	○	leading edge	15
unpleasant	○	○	○	○	○	○	○	pleasant	16
secure	○	○	○	○	○	○	○	not secure	17
motivating	○	○	○	○	○	○	○	demotivating	18
meets expectations	○	○	○	○	○	○	○	does not meet expectations	19
inefficient	○	○	○	○	○	○	○	efficient	20
clear	○	○	○	○	○	○	○	confusing	21
impractical	○	○	○	○	○	○	○	practical	22
organized	○	○	○	○	○	○	○	cluttered	23
attractive	○	○	○	○	○	○	○	unattractive	24
friendly	○	○	○	○	○	○	○	unfriendly	25
conservative	○	○	○	○	○	○	○	innovative	26

Figure 29: *English User Experience Questionnaire.*

Task Load Questionnaire (NASA TLX)

In my study I used the short version of the NASA TLX questionnaire, which is designed to compute the subjective workload introduced by a certain task [43]. I did not apply a weighting systems to the different subscales. The questionnaire was conducted separately for all thee authentication schemes, right after the User Experience Questionnaire.



Mental Demand

Low — High

How much mental and perceptual activity was required (e.g. thinking, deciding, calculating, remembering, looking, searching, etc)? Was the task easy or demanding, simple or complex, exacting or forgiving?

Physical Demand

Low — High

How much physical activity was required (e.g. pushing, pulling, turning, controlling, activating, etc)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?

Temporal Demand

Low — High

How much time pressure did you feel due to the rate of pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?

Performance

Good — Poor

How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?

Effort

Low — High

How hard did you have to work (mentally and physically) to accomplish your level of performance?

Frustration

Low — High

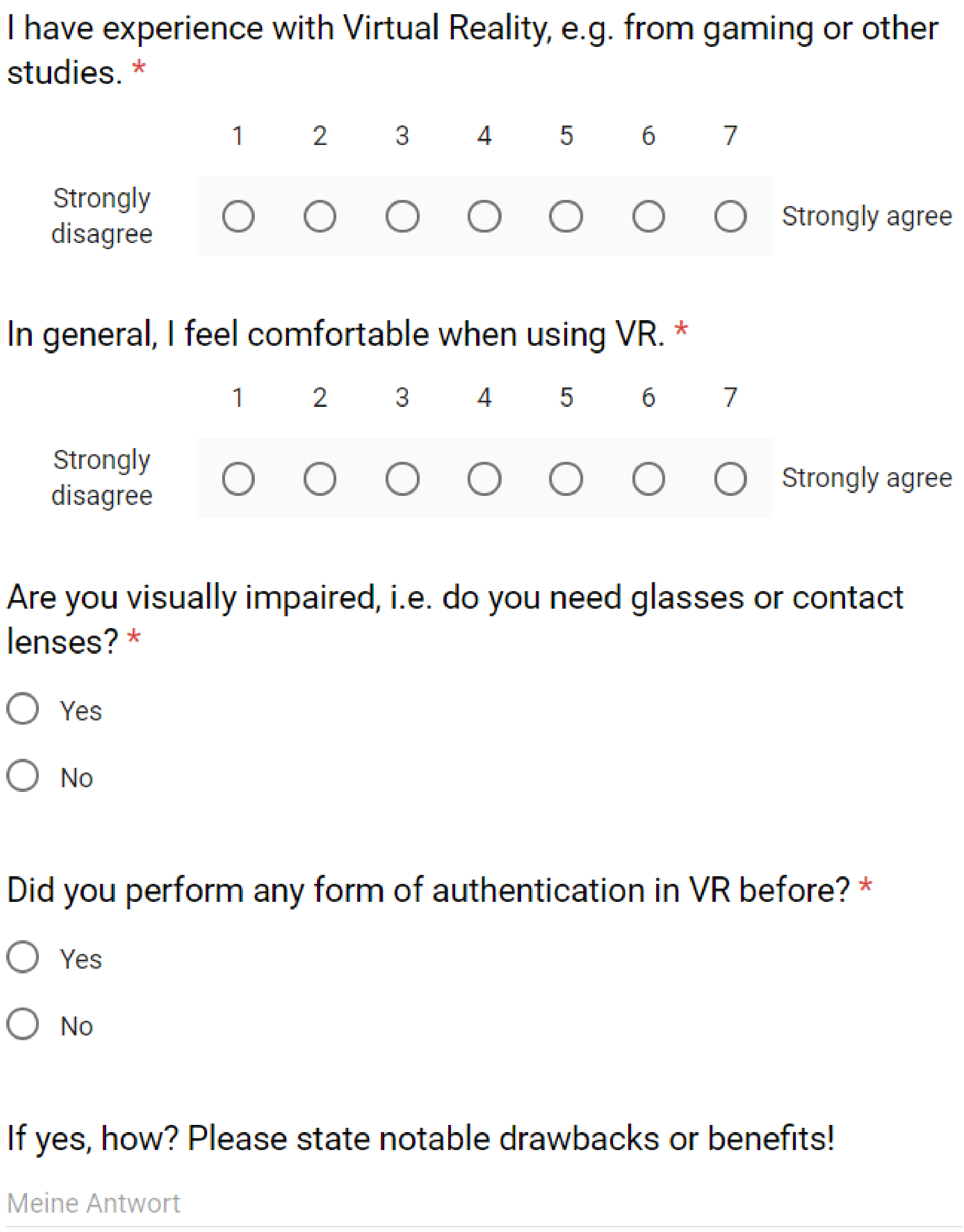
How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

Figure 30: *NASA TLX Questionnaire.*

Demographic Questionnaire

The last questionnaire that was answered by all participants of the study is a questionnaire I designed myself, which is designed to gather basic demographic information like e.g. age and gender. Additionally the questionnaire focused on gathering information about the experience the participant had with the three different authentication schemes and with Virtual Reality prior to the study. The participant was also asked to compare the three authentication schemes in the following metrics: is the scheme secure enough for me, how large would my password with this scheme be, how easy to memorize is a password with this scheme.

The entire questionnaire can be found on the following pages.



I have experience with Virtual Reality, e.g. from gaming or other studies. *

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

In general, I feel comfortable when using VR. *

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

Are you visually impaired, i.e. do you need glasses or contact lenses? *

○ Yes

○ No

Did you perform any form of authentication in VR before? *

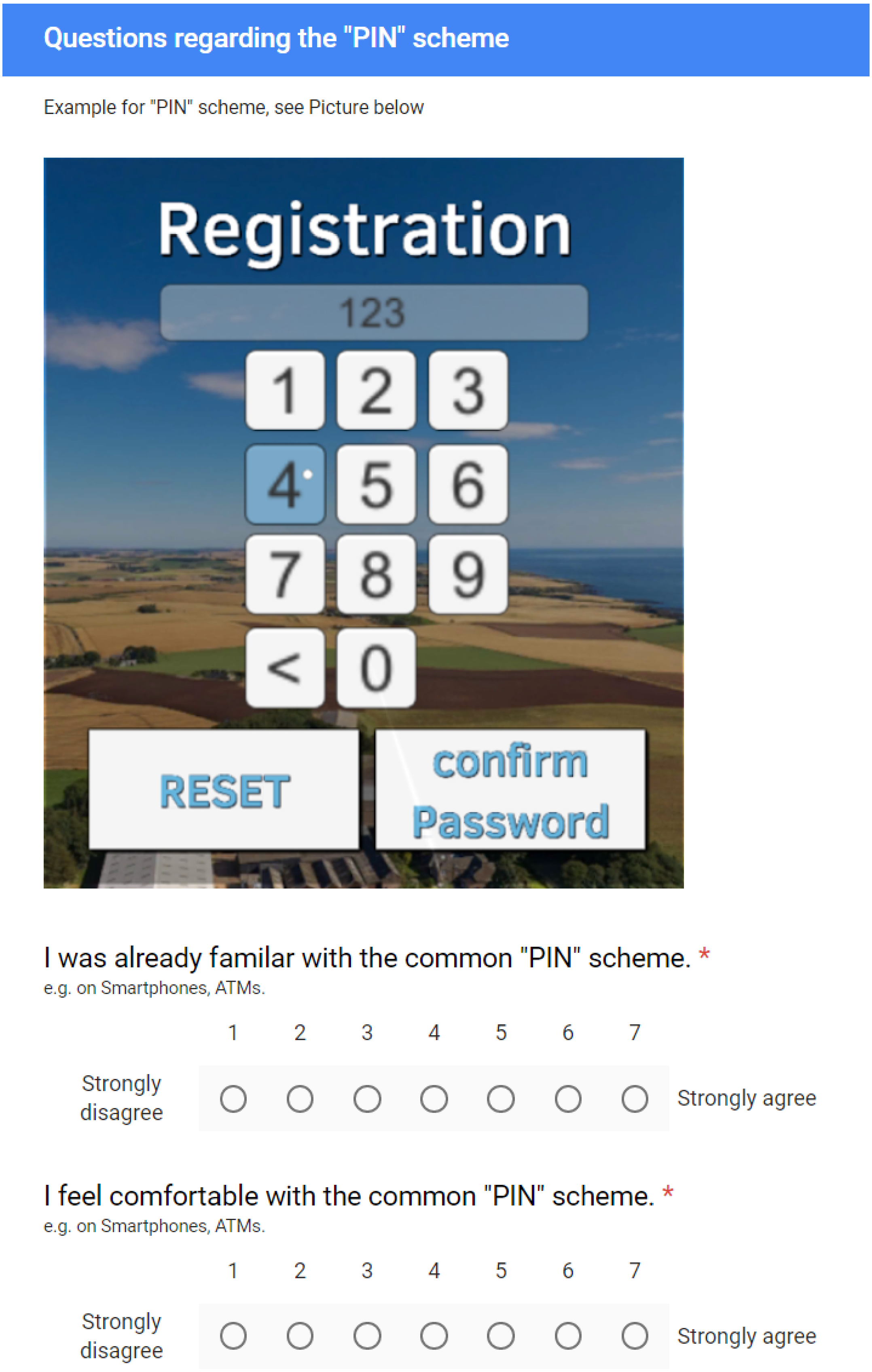
○ Yes

○ No

If yes, how? Please state notable drawbacks or benefits!

Meine Antwort

Figure 31: *First page of the demographics questionnaire.*



Questions regarding the "PIN" scheme

Example for "PIN" scheme, see Picture below

I was already familar with the common "PIN" scheme. *

e.g. on Smartphones, ATMs.

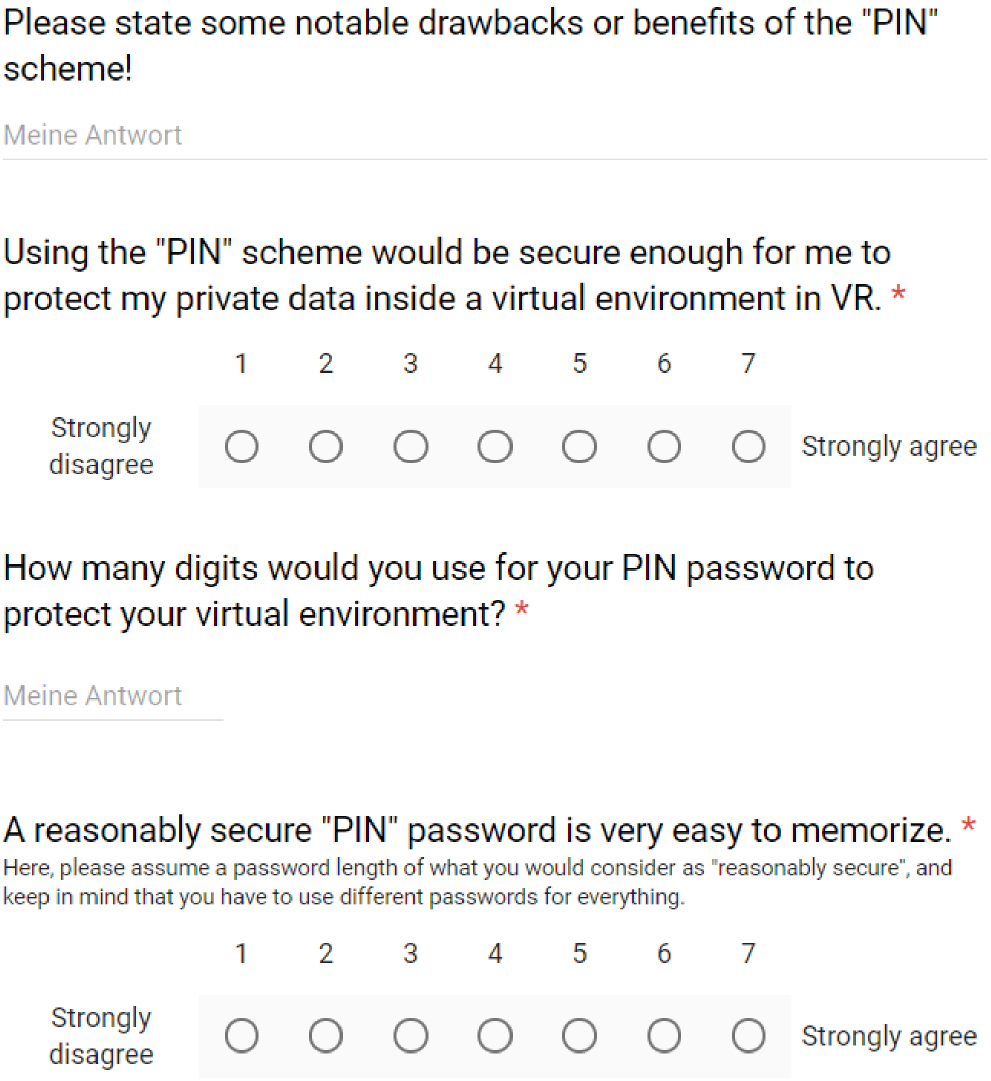
	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

I feel comfortable with the common "PIN" scheme. *

e.g. on Smartphones, ATMs.

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

Figure 32: *Second page of the demographics questionnaire.*



Please state some notable drawbacks or benefits of the "PIN" scheme!

Meine Antwort

Using the "PIN" scheme would be secure enough for me to protect my private data inside a virtual environment in VR. *

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

How many digits would you use for your PIN password to protect your virtual environment? *

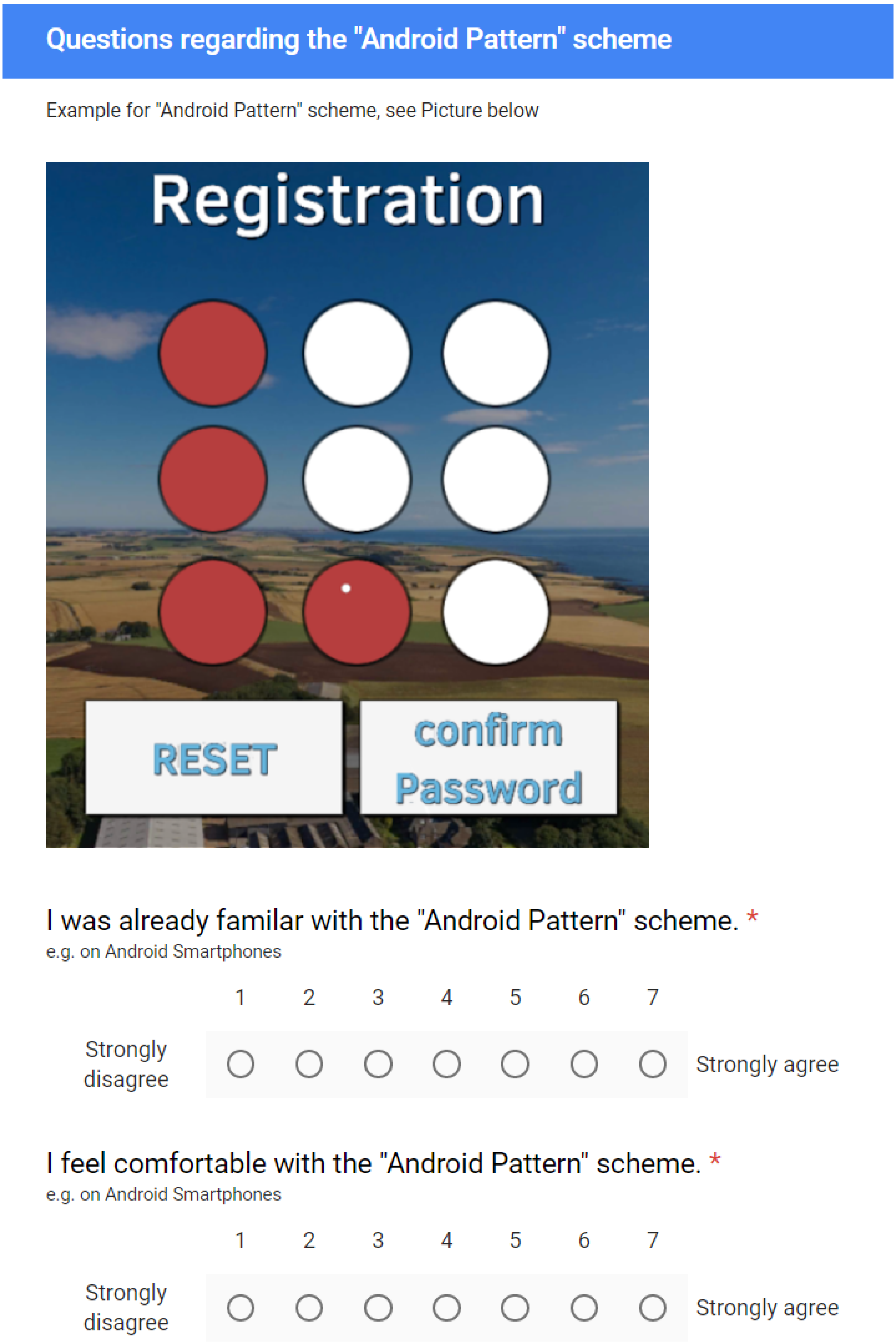
Meine Antwort

A reasonably secure "PIN" password is very easy to memorize. *

Here, please assume a password length of what you would consider as "reasonably secure", and keep in mind that you have to use different passwords for everything.

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

Figure 33: *Third page of the demographics questionnaire.*



Questions regarding the "Android Pattern" scheme

Example for "Android Pattern" scheme, see Picture below

I was already familar with the "Android Pattern" scheme. *

e.g. on Android Smartphones

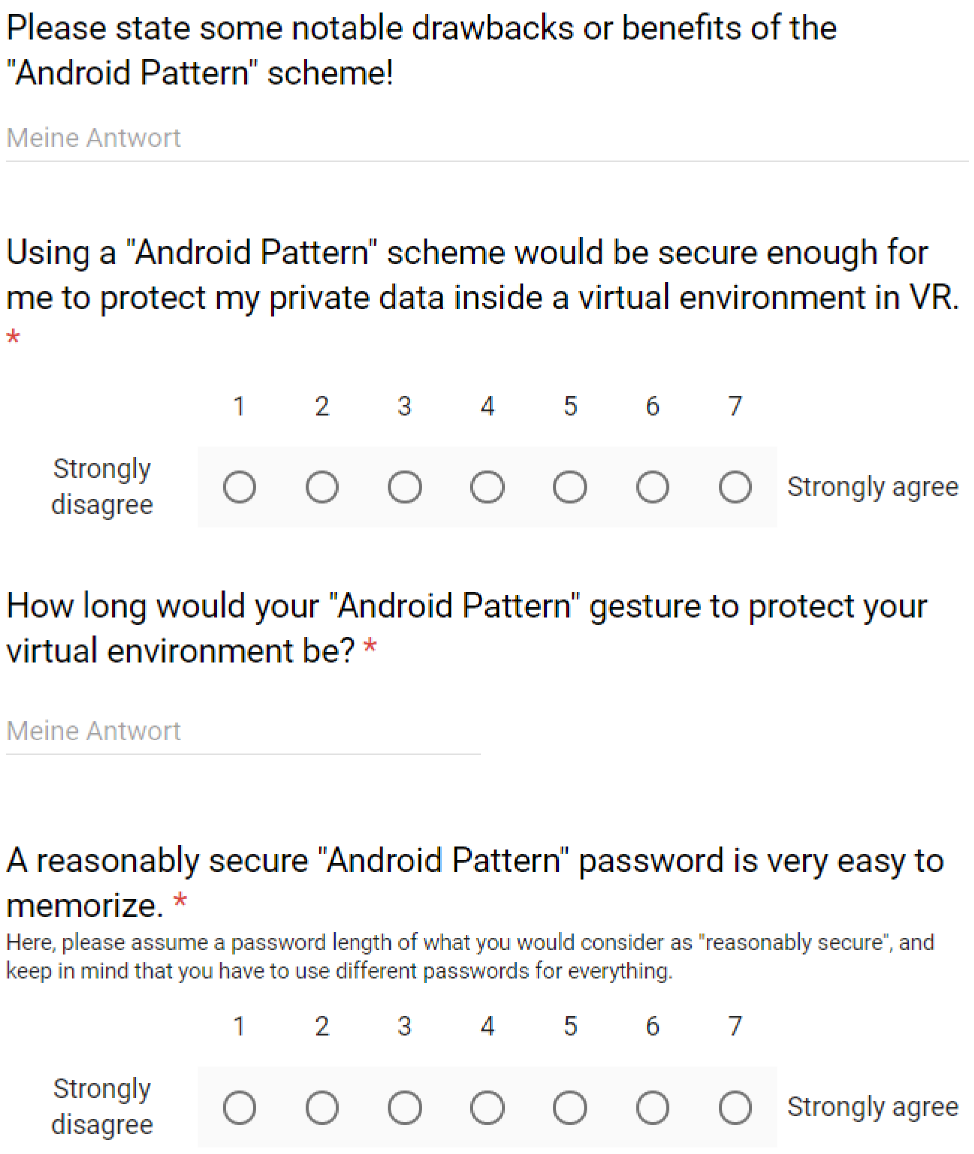
	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

I feel comfortable with the "Android Pattern" scheme. *

e.g. on Android Smartphones

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

Figure 34: *Forth page of the demographics questionnaire.*



Please state some notable drawbacks or benefits of the "Android Pattern" scheme!

Meine Antwort

Using a "Android Pattern" scheme would be secure enough for me to protect my private data inside a virtual environment in VR. *

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

How long would your "Android Pattern" gesture to protect your virtual environment be? *

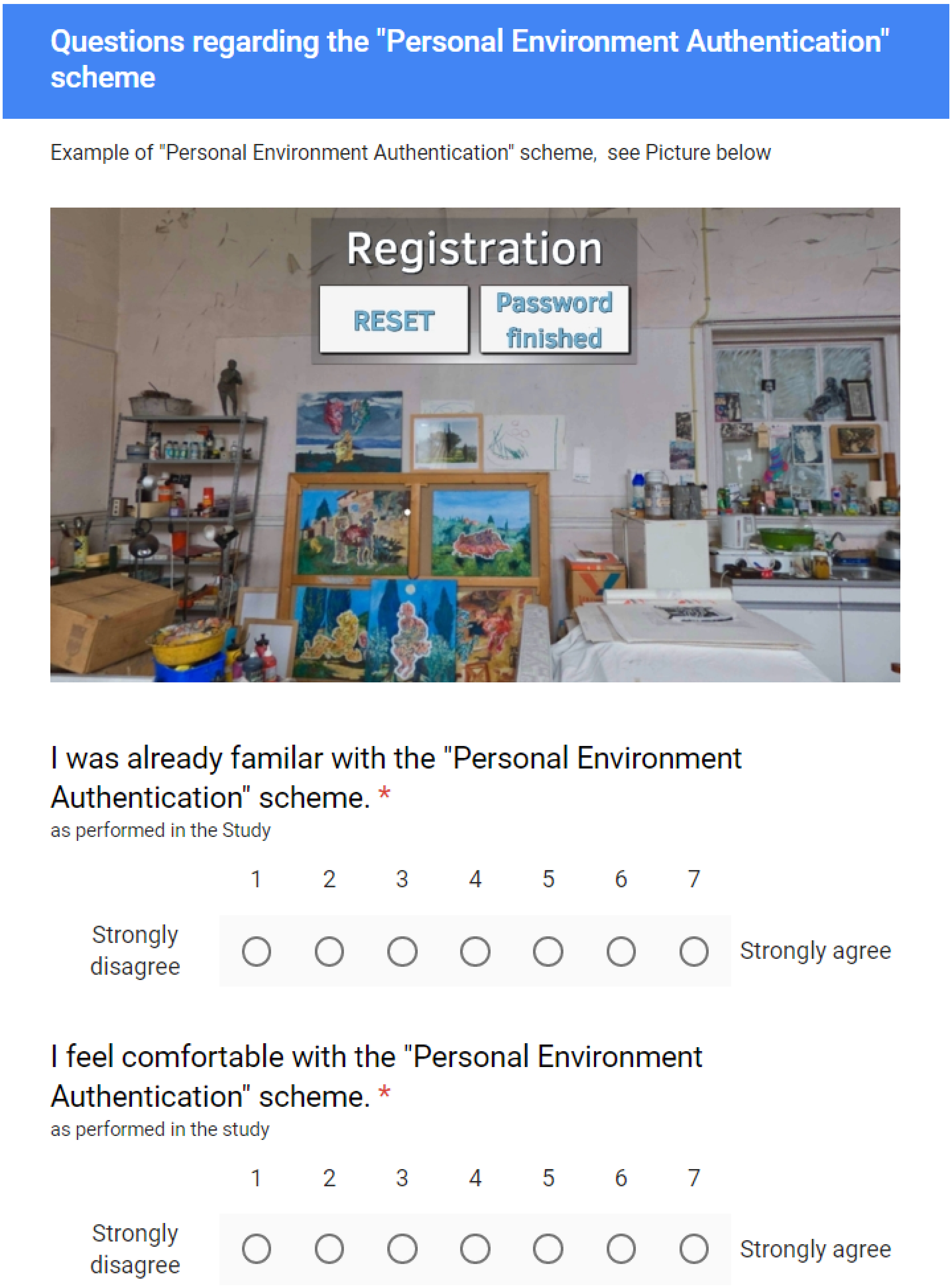
Meine Antwort

A reasonably secure "Android Pattern" password is very easy to memorize. *

Here, please assume a password length of what you would consider as "reasonably secure", and keep in mind that you have to use different passwords for everything.

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

Figure 35: *Fifth page of the demographics questionnaire.*



Questions regarding the "Personal Environment Authentication" scheme

Example of "Personal Environment Authentication" scheme, see Picture below

I was already familar with the "Personal Environment Authentication" scheme. *

as performed in the Study

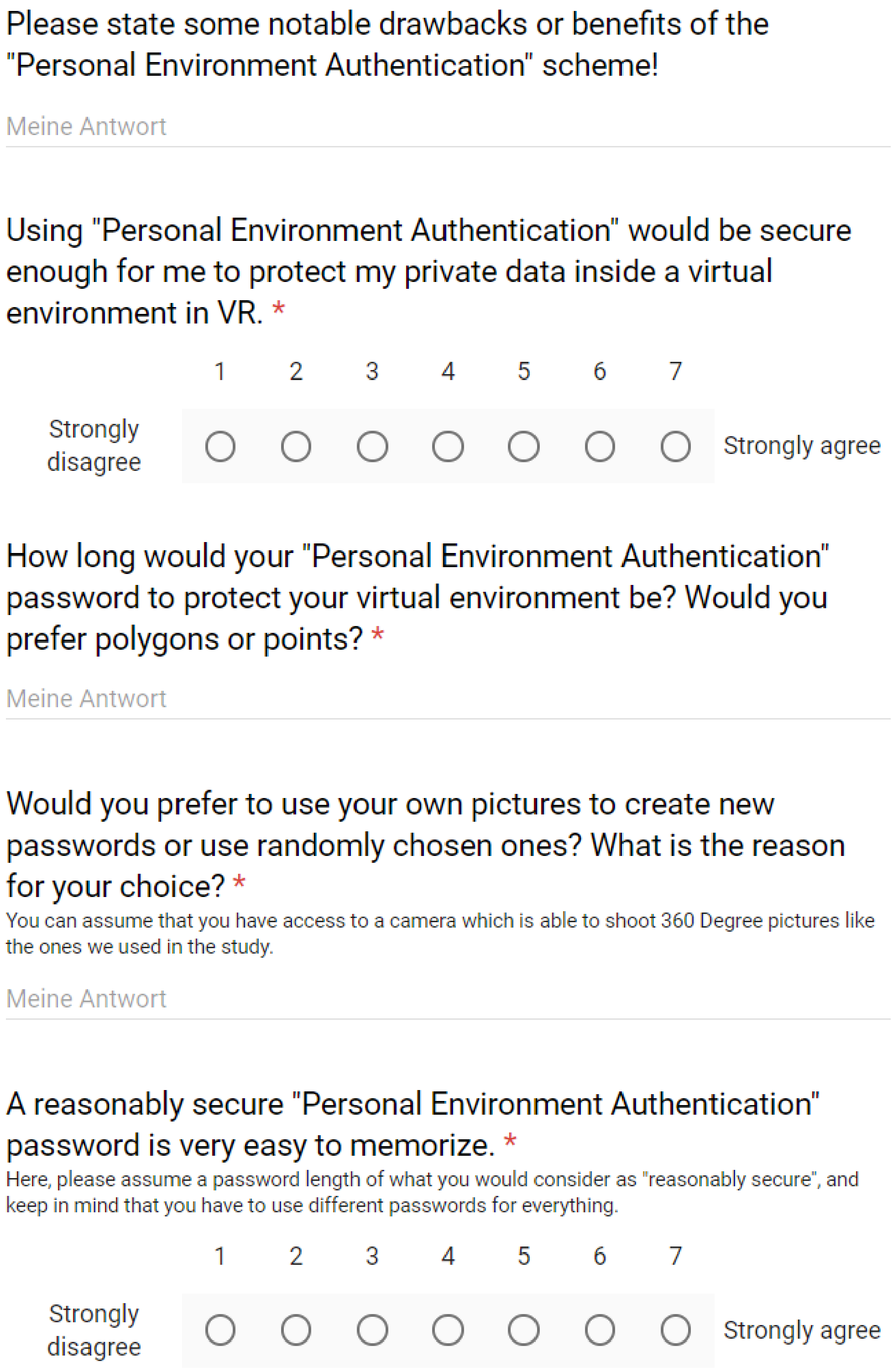
	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

I feel comfortable with the "Personal Environment Authentication" scheme. *

as performed in the study

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

Figure 36: *Sixth page of the demographics questionnaire.*



Please state some notable drawbacks or benefits of the "Personal Environment Authentication" scheme!

Meine Antwort

Using "Personal Environment Authentication" would be secure enough for me to protect my private data inside a virtual environment in VR. *

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

How long would your "Personal Environment Authentication" password to protect your virtual environment be? Would you prefer polygons or points? *

Meine Antwort

Would you prefer to use your own pictures to create new passwords or use randomly chosen ones? What is the reason for your choice? *

You can assume that you have access to a camera which is able to shoot 360 Degree pictures like the ones we used in the study.

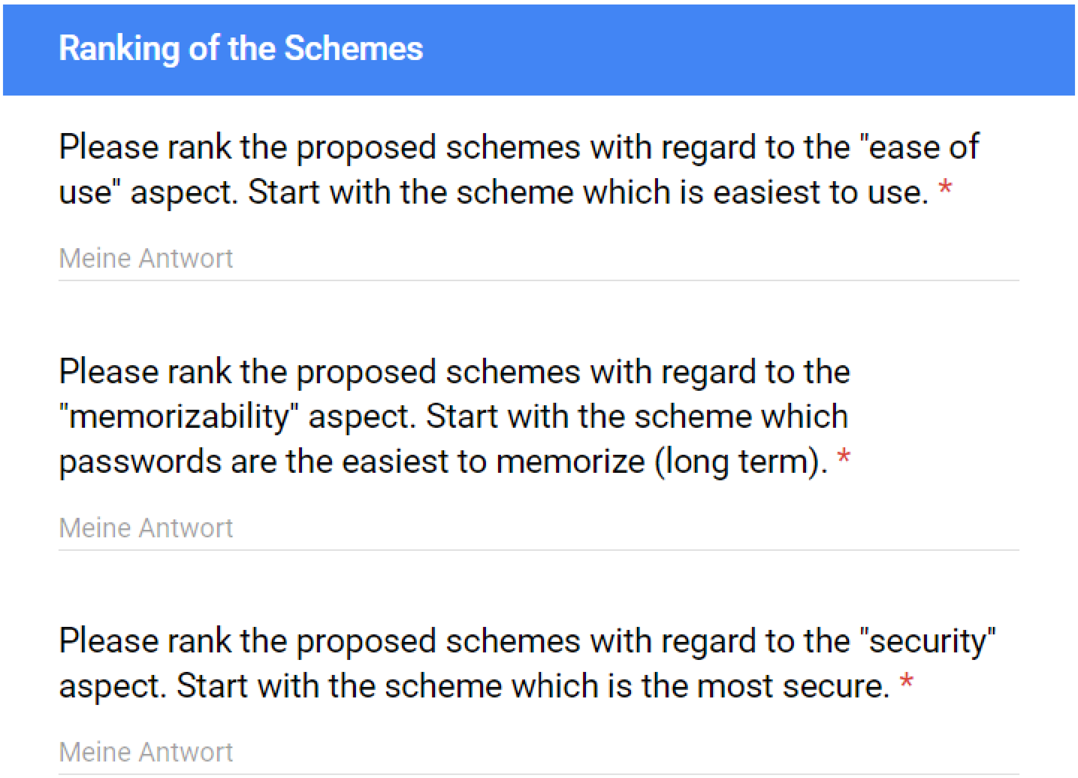
Meine Antwort

A reasonably secure "Personal Environment Authentication" password is very easy to memorize. *

Here, please assume a password length of what you would consider as "reasonably secure", and keep in mind that you have to use different passwords for everything.

	1	2	3	4	5	6	7	
Strongly disagree	○	○	○	○	○	○	○	Strongly agree

Figure 37: *Seventh page of the demographics questionnaire.*



Ranking of the Schemes

Please rank the proposed schemes with regard to the "ease of use" aspect. Start with the scheme which is easiest to use. *

Meine Antwort

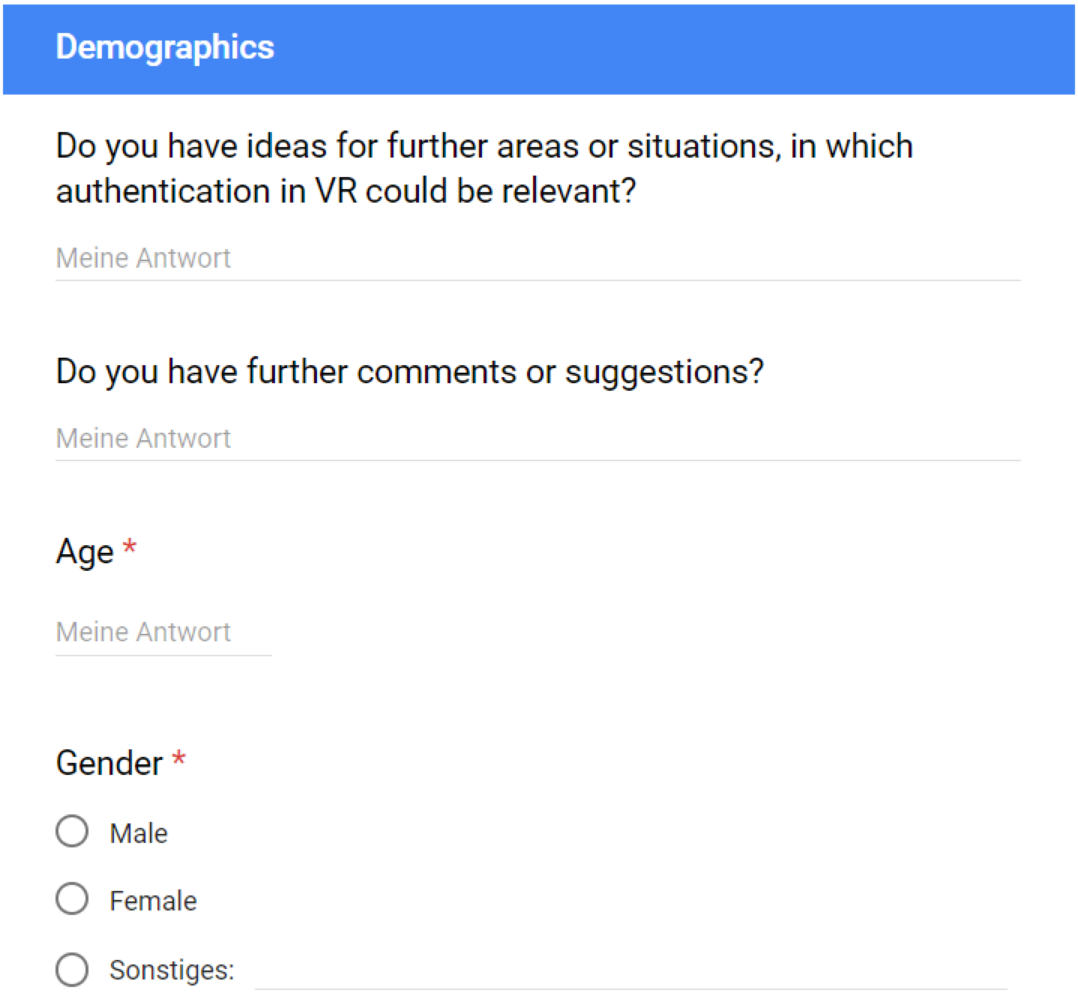
Please rank the proposed schemes with regard to the "memorizability" aspect. Start with the scheme which passwords are the easiest to memorize (long term). *

Meine Antwort

Please rank the proposed schemes with regard to the "security" aspect. Start with the scheme which is the most secure. *

Meine Antwort

Figure 38: *Eighth page of the demographics questionnaire.*



Demographics

Do you have ideas for further areas or situations, in which authentication in VR could be relevant?

Meine Antwort

Do you have further comments or suggestions?

Meine Antwort

Age *

Meine Antwort

Gender *

○ Male

○ Female

○ Sonstiges:

Figure 39: *Ninth page of the demographics questionnaire.*