SCARII: 증강 현실 환경에서 의도하지 않은 입력으로 인한 안전 문제를 방지하는 방법에 대한 연구

SCARII: A Study to Advise Safety Concerns Caused by Involuntary Inputs in Augmented Reality

채한주, 황정인, 김이은, 고 건, 서진욱[1)]
Han Joo Chae, Jeong-in Hwang, Yieun Kim, Kyle Koh, Jinwook Seo

(08826) 서울시 관악구 관악로 1 서울대학교 컴퓨터공학부
{hanjoo, jihwang, yekim, kyle}@hcil.snu.ac.kr, jseo@snu.ac.kr

요 약

본 논문에서는 증강 현실 환경에서 각종 IoT (Internet of Things) 기기들을 조작하면서 발생할 수 있는 다양한 안전 문제를 탐색하고, 이를 방지하는 디자인 가이드라인을 제시한다. 먼저 의도치 않은 갑작스러운 변경을 방지하고 사용자가 원하는 값으로 조절할 수 있도록 하는 보호 인터페이스를 디자인하고, 이를 mid-air 터치 및 손목 회전 제스쳐에 적용한 사용자 실험을 통해 본 인터페이스의 사용성과 안전성을 검증한다. 또한, 가전 제품들을 사용자가 직접 조작하도록 하는 관찰 연구를 진행하여 다양하고 실제 발생 가능한 안전 관련 이슈들을 탐색한다. 연구 결과로 사용자들이 갑작스러운 변화로 인한 위험도가 높을수록 본 연구에서 제안하는 인터페이스를 선호함을 밝히고, 정성적 분석을 통해 안전 관련 문제를 분류하여 이에 기반한 디자인 가이드라인을 제안한다.

Abstract

In this paper, we explore various safety hazards that can be caused by involuntary inputs while configuring IoT (Internet of Things) devices in Augmented Reality (AR) environment and propose design guidelines to advise such risks. As a first step, we designed a safeguard interface which sets pre-conditions to prevent accidental changes while allowing users to deliberately adjust the parameters to their desired values. We evaluated the usability and safety of our interface designed with mid-air touch and wrist-rotation gestures through a controlled user study. We also conducted an observational study in a realistic setting where users controlled home appliances using our interface in order to explore various safety related issues. The results have revealed that the proposed interface was preferred when the configurational changes can cause more critical or irreversible damage. Also, through the qualitative analysis, we categorized safety related issues and suggested design guidelines based on the results.

1) 교신저자

논문투고일자 : 2020. 02. 26 심사일자 : 2020. 03. 13 게재확정일자 : 2020. 04. 21

키워드: 의도하지 않은 입력, 안전 장치 메커니즘, 증강 현실, IoT
Keyword: Involuntary Input, Safeguard Mechanism, Augmented Reality, Internet of Thing

1. Introduction

With the recent advance in Internet of Things (IoT) and Augmented Reality (AR) technologies, it is not surprising to remotely configure connected devices, such as the volume of speakers, the brightness of lamps, or the temperature of heaters, using mid-air gestures [1] as well as dashboards on mobile, PC or wearable devices [2-5]. Mid-air gestures, in particular, possess greater potential in AR environment because users can remotely configure the devices at sight in a quick and efficient manner without having to occupy hands for supplementary controllers [1]. However, such convenience can often be a double-edged sword due to the safety problems it can embrace such as damaging ears by accidentally raising the volume of a speaker to its maximum. Similar scenarios are conceivable with lamps, water temperature, or gas stoves. As configurational changes will directly affect users' surroundings, safety concerns become more critical when controlling devices in AR environment.

Even with the tremendous progress in their accuracy, current state-of-art gesture recognition techniques are still far from perfect due to their inability to reliably detect and filter out all human errors such as diverted attention and misjudgments that cause involuntary inputs. Especially, when configurational changes to devices are immediately reflected to their functions and the surroundings, even a single false-positive could result in devastating consequences. Although many system level approaches to filter out unimportant touch inputs [2, 6] and mid-air gestures [7-8] were presented in the past, the safety issues caused by unintended configurational changes using mid-air gestures in AR environment are yet to be fully explored. We define such problems in an umbrella term, SCARII (Safety Concerns in Augmented Reality environment caused by Involuntary Inputs).

In order to elaborate on how to address SCARII, we first focused on the mistakes caused by the human's diverted attention or the system's recognition errors based on the previous studies described in the following section. We then designed a safeguard interface which sets pre-conditions to prevent abrupt changes when involuntary input occurs. To evaluate its usability and safety, we conducted a controlled user study that simulates device configuration by having users change values on the floating meters using our interface. We found that our interface supported reducing unintentional inputs made by the users and they were willing to trade-off time performance for safety. As we wanted to explore more in SCARII, we also conducted an observational study in a realistic setting where users controlled home appliances using the same interface. After qualitative analysis of the results, we categorized each safety related issues and developed design guidelines that address problems in each category to avoid SCARII issues.

The contributions of this paper are as follows:

1) We presented the design of a safeguard interface that helps users avoid making involuntary inputs using mid-air gestures in AR environment.

2) We applied our interface to mid-air touch and wrist-rotation gestures and evaluated the usability and safety of our interface with a controlled user study.
3) We conducted an observational study in a realistic setting with real-life scenarios to explore various safety issues.
4) We presented different categories of SCARII and the design guidelines which can be used when developing safe AR interfaces.

2. Related Work

We give a brief summary of the existing techniques relevant to the design of safeguard interfaces in AR environment.

2.1 Augmented Reality and Device Configuration

In order to control and configure devices in a connected world, there are both virtual and physical approaches. Many works rely on virtual instance of objects using mobile-based AR techniques [9-16], and AR techniques on Optical (see-through) HMDs or Virtual Reality headsets with front cameras to emulate the 'AR' aspects of Optical HMDs [1, 17-18]. Also, some methods utilize proxy objects as physical intermediaries or avatars in order to control other devices in inconvenient or unreachable regions like ceiling and remote locations [16, 19-21]. Moreover, though they are not directly targeted to be used in AR environment, there have been various gesture based approaches to configure devices [22-25]. Having all these studies related to remote AR interactions, there is few prior work dealing with safety-aware interactions in AR space. Our current prototype used in the user experiments utilizes mid-air gestures as primary input methods and keeps the see-through concept of AR by emulating it with a VR headset and a set of cameras attached in front, projecting the exterior view to the screen. The same interaction techniques can easily be implemented in AR headsets with an Optical HMD.

2.2 Avoiding Involuntary Inputs or Human Errors

Although they are not specifically designed for AR space, a number of studies have explored the ways to systematically distinguish accidental inputs [8]. In addition, to enhance the general quality of interaction, there have been various approaches to filter out unimportant gestures such as involuntary finger or palm touches on mobile and tablet devices [2, 6, 26], as well as unintentional hand motions including wrist rotation and other mid-air gestures [7, 27-28]. Our interface and the design guideline are specifically designed to address SCARII when mid-air gestures are used to configure IoT devices.

Also, human errors are not negligible when it comes to the safety issues; they are rather critical. If a user mistakenly changes configuration to the undesired level or forgets the correct procedure, unintended change will occur regardless of the integrity of the gesture recognition. For this matter, Embrey [29] divided human failure into errors and violations and then classified errors once again into three different types: skilled-based slips, rule-based mistakes, and knowledge-based mistakes. Our safeguard features can effectively handle safety issues caused even by these human errors and we later adopt this classification of human errors to categorize SCARII and build design guidelines.

3. Safeguard Interface

To design a safeguard interface which sets pre-conditions to prevent accidental changes while allowing users to deliberately adjust the parameters to their desired values, we first investigated various types of mistakes that can easily occur during the configuration of IoT devices using mid-air gestures in AR environment. Based on the systematic approaches [8] and human errors [29] described in the related work, we identified when the SCARII problems may occur while configuring devices: i) when a user mistakenly slips to select unintended location on the meter ii) when a user makes a mistake due to one's diverted attention or misjudgment, and iii) when a system erroneously thinks a certain location on the meter, where a user has not intended, has been selected.

3.1 Selection Mechanism

We adopted a dragging-like selection mechanism that is commonly used for sliders [30] where a user is required to touch-and-hold (*select*) while making changes (*change*). Once the value reaches the user's desirable value, s/he can simply let go of the touch (*deselect*). As a touching action demonstrates a clear intention for adjustment, unnecessary noise before and after the configuration can be eliminated.

3.2 Immediate vs Delayed Update

There are two different ways to update the changes made by mid-air gestures to the object: immediate and delayed. The former is useful for observing volume or brightness changes, while it can be annoying when accidental changes are made. On the other hand, the latter is more error-safe and more adequate when there is no need for a closed-loop feedback such as changing temperature of a thermometer. During the study, we focus on exploiting the strengths of the immediate update method while alleviating the result of unintentional inputs.

3.3 Safeguard Features

In order to advise the identified SCARII problems, we adopted and generalized the previous design of four different safeguard mechanisms [31] in addition to the baseline method with no safeguard feature. The GUI Fig. 1(a) consists of two components: a cursor that always show the current position of a user's hand whether or not it is *selected* and the current value of a setting.

3.3.1 Absolute (ABS) - Baseline

A user's hand position is absolutely mapped to both the cursor and the gauge of the meter and any change on the hand position is immediately applied until s/he *deselects* it. As no safeguard feature is provided in the *ABS*, accidental changes may occur when the user's initial hand position upon *selection* is different from the current state.

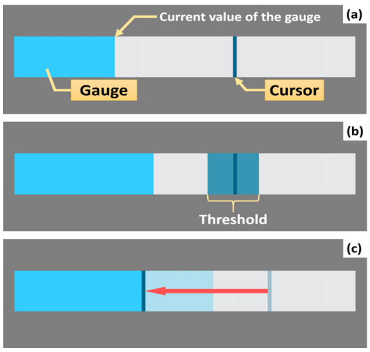


Fig. 1 GUI of the safeguard interface

3.3.2 Absolute-Threshold (ATH)

A threshold is added to the *ABS* and it requires a user to move a certain distance greater than the threshold from where s/he *selected* in order to start making changes. This would give the user a room to cancel by *deselecting* before applying any changes in cases the *selection* was done in unintended hand position or by accident. The amplitude of the threshold is visualized on the meter as shown in Fig. 1(b).

3.3.3 Absolute-Magnet (AM)

AM requires a user to *select* and move the cursor to the current value of the gauge before the gauge starts to move (Fig. 1c, along the arrow). Once the cursor meets the gauge (magnetized), the gauge will follow the cursor just like *ABS*. As such mechanism requires a user to deliberately choose the current value and before making changes, it is expected to prevent sudden or accidental changes.

3.3.4 Absolute-Trail (AT)

Similar to *AM*, a user may *select* and move the cursor to the current value of the gauge. However, until the cursor and the gauge are met, the gauge also slowly trails towards the cursor’s position automatically. This results in bridging the distance between them, while preventing any sudden changes. Once they meet, the gauge no longer slowly trails the cursor; it immediately follows the cursor. Therefore, instead of deliberately snatching the gauge like *AM*, a user can also just hold the cursor on top of the desired location and wait until the gauge to arrive. This is similar to how we hold the +/− button on a remote controller and wait for the volume to reach the desired level.

3.3.5 Relative (R)

Unlike other methods, *R* uses a relative mapping. The value of the gauge changes only for the distance the hand has moved while the cursor is *selected* regardless of the hand’s starting position.

3.4 Prototype Design

In order to evaluate our safeguard interface as well as explore SCARII issues, we developed a prototype which enables users to interact with IoT devices in AR environment.

3.4.1 Video-based AR with Gesture Detection

To keep the see-through concept of AR, we attached two webcams (Logitech C270) to a VR headset (Oculus Rift DK2) as shown in Fig. 2. In addition, we mounted Leap Motion on top of the headset for hand-gesture recognition.

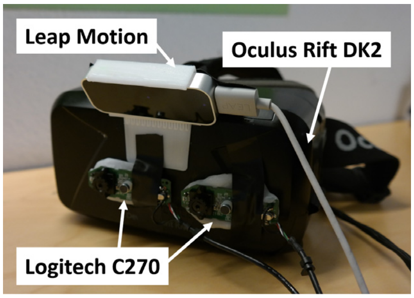


Fig. 2 A prototype of the AR HMD

3.4.2 Choices of Mid-air Gestures

Among the popular mid-air gestures [32], we filtered out those that are inadequate for device configuration and chose the two most popular ones: mid-air touch and hand-wrist rotation. For mid-air touch, a user touches the meter Fig. 3(a) with her/his index finger and changes the gauge value by moving left or right. For wrist rotation, s/he rotates her/his wrist to change the gauge value on the meter Fig. 3(b). The arc-shaped radial meter Fig. 4 was used specifically for the wrist rotation because such visual feedback is important for users to understand the range of motion [33]. For fair comparison, the average length of the outer and the inner arc was matched to the length of the linear meter and the usage mechanisms of both meters were identical.

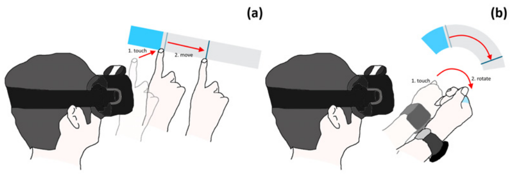


Fig. 3 Mid-air touch and wrist rotation

3.4.3 Gesture Recognition Methods

In our prototype, different recognition methods were used for each gesture.

Mid-air touch: Both the selection and horizontal movement are recognized by Leap Motion mounted on the headset Fig. 3(a). To be more specific, a user can select the cursor by placing the index finger on top of the meter that appeared 30cm in front of her/him and then drag the gauge value by moving left or right while touching. S/he can simply release the touch to finish the modification.

Wrist rotation: The selection gesture is detected by a ring-type capacitive sensor worn on the user's finger and the rotation gesture is recognized by a smartwatch (Galaxy Gear S) worn on the user's wrist Fig. 3(b). That is, s/he can touch the ring to select the meter and then rotate one's wrist to adjust the gauge value. Releasing the touch also works as an indicator of finishing the configuration. The sensor-based detection method was used because vision-based detection was unstable with wrist rotation and we wanted to focus solely on the SCARII and eliminate other noises, such as inaccurate gesture recognition.

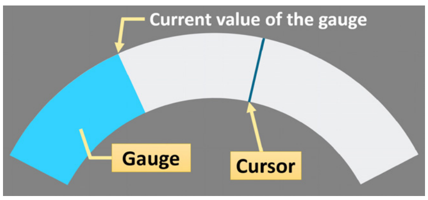


Fig. 4 Arc-shaped radial meter

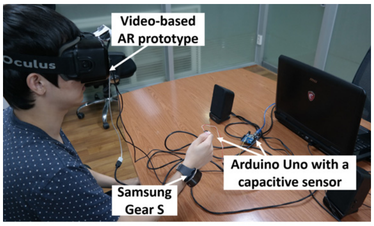


Fig. 5 Setup of the controlled user experiment

4. Controlled user experiment(study 1)

We conducted a controlled user study to evaluate usability and safety of our safeguard features. In terms of usability, we were interested in how quickly, accurately, and conveniently the participants change the value to a set target. As for safety, we wanted to evaluate how effectively each feature handles involuntary inputs during the tasks.

4.1 Participants

We recruited 12 participants (10 men and 2 women) between the ages of 19 and 28 (μ=24, σ=3) from a local university. All participants had normal or corrected-to-normal vision (20/25) and were right-handed. Only one participant had previous experience with immersive HMDs for less than 7 days and none of them had any experience with smartwatches.

4.2 Apparatus

Our prototype described in the previous section was used to visualize and control the meter using mid-air gestures. The application was developed on a PC using Unity 5 and SDKs for Oculus Rift and Leap Motion. For the wrist-rotation gesture, Gear S was connected through a wireless router and the ring-type capacitive sensor was connected to an Arduino Uno board which was connected to the PC via USB. The overall setting of the experiment is shown in Fig. 5.

4.3 Task

The participants were asked to interact with a floating meter that appeared 30cm in front of them. The goal was to move the gauge value into the red target range Fig. 6 as quickly and accurately as possible. The initial gauge value and the position of the target were randomly selected for each task. The length of the meter and the width of target range were set to 35.58cm and 5.08cm [34] respectively.

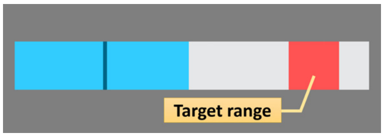


Fig. 6 Linear meter with a target range

4.4 Measures

The goal of the experiment was to verify that our safeguard features were effective without significantly damaging the usability of the inter−face. We measured completion time to quantita−tively evaluate the general performance and SUS scores [35] and rankings of the participants' preferred interfaces to qualitatively assess the usability and preference of the interface.

4.5 Procedure

The experiment was a within−subjects design with 2 gestures (mid−air touch and wrist rotation) and 5 safeguard mechanisms (*ABS*, *ATH*, *AM*, *AT*, and *R*) as independent variables, resulting in 10 different conditions. The order of the gestures was counterbalanced and the order of the safeguard mechanisms was randomized for each gesture.

The participants performed the gauge confi−guration tasks for the 10 different conditions. Before each condition, the participants were informed about the usage and allowed to practice until they felt comfortable. Each participant performed 10 trials per condition, which resulted in total of 100 trials per participant (2 gestures × 5 safeguard mechanisms × 10 trials/condition). The completion time was measured for each task and the participants were asked to fill out SUS after each condition. We did not explain the real purpose of each of the five different mechanisms until the end of the experiment, and during the tasks, the participants only knew we were interested in the time performance and the usability. At the end of the experiment, we asked each participant to rank the 5 different mechani−sms based on the usability preference. Then, we explained about the safety concerns and the purpose of each safeguard feature designed to advise SCARII and asked each participant to re−rank the mechanisms based on safety. Lastly, we asked them to re−rank them considering both usability and safety. We also interviewed the participants in between each step.

4.6 Results

4.6.1 Task Completion Time

A two−way repeated measures ANOVA [36] was conducted to examine the effect of gestures and safety features on completion time. There were statistically significant main effects of both ges−tures ($F_{1,11}=40.1$, $p<.001$) and safety mechani−sms ($F_{4,44}=6.33$, $p<.001$) on completion time, while no significant interaction was found. In terms of gestures, Bonferroni post−hoc analysis has revealed that the mid−air touch was significantly slower than the wrist rotation ($p<.001$). In terms of safety mechanisms, *ABS* was significantly faster than *AM* ($p<.023$), *AT* ($p<.023$), and *R* ($p<.011$). However, there was no significant differences among other safeguards. The detailed descriptive statistics of the results are shown in Fig. 7.

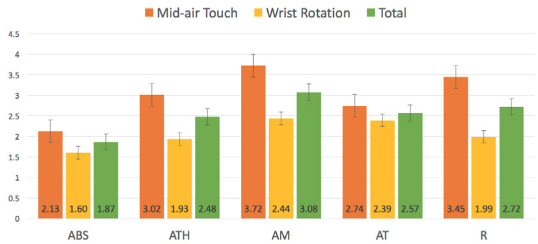


Fig. 7 Descriptive statistics of the completion time (in seconds) from Study 1

4.6.2 SUS Scores

The average SUS scores were calculated for each gesture-and-safeguard combination Fig. 8 and revealed that *ABS* and *AT* scored above average (> 68) with both gestures indicating that they were fairly easy to use and learn [37] regardless of the gestures used.

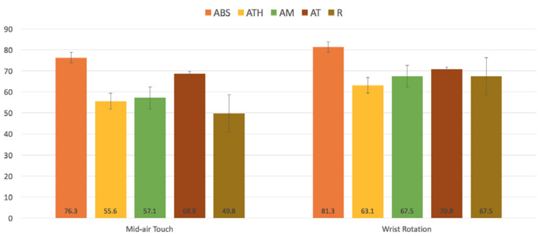


Fig. 8 Average SUS scores from Study 1

4.6.3 Rankings and Interview

At the end of the experiment, we collected user rankings of the safeguard mechanisms based on three different aspects: usability, safety, and overall Table 1. Friedman tests were conducted for each of the three aspects. For usability, there was a statistically significant difference on the ranking, $x^2(4) = 34.13$, $p < .001$, and post hoc analysis of Wilcoxon signed-rank test with Bonferroni correction revealed that *ABS* was significantly better than *ATH, AM,* and *R* ($p < .001$). There was no significant difference between *ABS* and *AT*. For safety, there was a statistically significant difference, $x^2(4) = 40.03$, $p < .001$, and the post hoc analysis revealed that *ABS* was significantly unsafe than all the other mechanisms ($p < .002$). For overall, there was a statistically significant difference, $x^2(4) = 15.63$, $p < .004$, and the post hoc analysis revealed that *AT* was significantly preferred over *ABS* when users consider both usability and safety.

Table 1. Average rankings of the safeguard mechanisms from Study 1 (lower number indicates better ranking). Winners of each aspect are highlighted.

	ABS	ATH	AM	AT	R
Usability	1.50	3.50	3.50	2.67	3.83
Safety	4.67	2.54	1.92	2.92	2.96
Overall	3.79	2.67	2.54	2.42	3.58

During the interview, it was interesting to observe the changes in each participant's attitude towards the safety features after we explained about SCARII. Except for one participant who anticipated the concern and said, "*R* would fit well with volume control (P1 from Study1, S1P1)" even before we told him about the purpose of the study, most participants did not conceive any SCARII and preferred *ABS* the most for its efficiency. However, as we explain the purpose of the safety features, preferences towards *ABS* critically declined as a number of participants reported that *ABS* would cause unintended results due to its nature of immediate update (S1P1, S1P3, S1P11, and S1P12). On the other hand, the rank of *AT* which scored above average in SUS and ranked the second place in terms of usability, was little affected in safety and even scored the first place in overall preference. The ranking results were

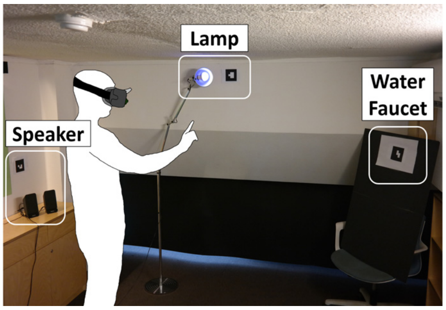


Fig. 9 Setup of the observational user experiment

consistent with the interview as some of the participants reported that they liked *AT* for both usability and safety (S1P7, S1P8, and S1P12), that it was convenient to just wait for the gauge to come (S1P7, S1P9, and S1P10), and that they had enough time to recover by observing the visual feedback (S1P2, S1P4, and S1P11).

Based on the verdict, we found that *ABS*, without any safeguard feature, was efficient to use, yet less preferable when there is SCARII. We also found that users generally chose safety at the cost of time and *AT* provided well balanced interface with both usability and safety. In other words, although *AT* may not be as responsive as *ABS*, it would be more suitable for general purposes.

In terms of gestures, there were few minor complaints about the relative recognition accuracy of the mid-air touch gesture, but it was not critical since all participants still used the gesture well enough and it was quite obvious for the Gyro-sensor-based wrist rotation detection to be more precise compared to the vision-based touch gesture.

5. Observational user experiment(study 2)

The findings from the Study 1 revealed the trade-off relationship between time performance and safety, and *AT* was demonstrated to provide a satisfying level of both usability and safety. However, our safeguard interface does not seem to cover the entire domain of SCARII in device configuration. In order to extend our understanding about real-life safety concerns, we designed an observational study in a realistic setting where users configured home appliances using our interface in AR environment.

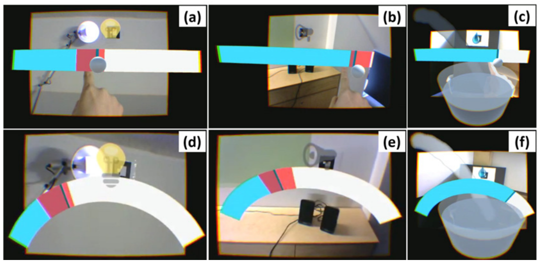


Fig. 10 A user's point-of-view when controlling a lamp, a speaker and a water faucet

For this study, in addition to the baseline *ABS*, we used the two safety mechanisms, *AM* and *AT*, which performed the best in the safety and the overall criteria respectively. The identical gestures, mid-air touch, and wrist rotation as Study 1, were used in this study.

Three devices, a lamp, speakers, and a water faucet with a bucket underneath, were placed in a room in a way that users can walk and look around to control them Fig. 9. While the other two were real physical devices, the water faucet was virtually simulated due to mechanical limitations of water supplies and difficulty of cleaning up the spills for every task Fig. 10. The devices were recognized by the camera mounted on the HMD using the AR markers placed near each device. When a marker was detected within the sight, a small icon appeared on top of the marker to indicate that the device was ready to be configured. Finally, when the icon was within 25 degrees of field of view from the center (that is, users moved their heads to place the icon at the center of the view), an adjustable meter appeared, similar to Study 1 Fig. 10.

5.1 Participants

12 participants (9 men and 3 women) between the ages of 23 and 28 (μ=25, σ=1) were recruited from a local university. All participants had normal or corrected-to-normal vision (20/25) and two participants were left-handed. However, none of them had difficulty using their right hands for gesture interactions. Two participants had previous experience with smartwatches, but none with AR headsets.

5.2 Apparatus

An identical HMD as well as gesture detection methods as the first study was used. For the lamp, Smartluce's Magic Bulb was controlled via Bluetooth LE. Also, Creative Labs' A60 speakers were connected to and controlled by the PC.

5.3 Tasks

There were three different tasks controlling each device.

5.3.1 Lamp

This task was to change the brightness of the lamp. The left most point of the gauge turned it off, while the right most point set it to the maximum brightness. When each task began, its initial brightness was randomly set and participants needed to adjust the gauge to the red target marked on the meter Fig. 10(a), (d).

5.3.2 Speakers

Similar to the *Lamp*, this task was to change the volume of the speakers. The left most point of the gauge muted the speakers while the right most point set it to the maximum volume. When each task began, their initial volume was randomly set and participants needed to adjust the gauge to the red target marked on the meter Fig. 10(b), (e).

5.3.3 Water Faucet

This task was different from the other two, not only because it was a virtual simulation, but also it did not have a red target marked on the meter Fig. 10(c), (f). The participants instead had to fill up the bucket on the chair by controlling the rate of the water pouring out from the faucet. This was designed to simulate a more complex and realistic task of filling the bucket quickly as well as accurately by adjusting the gauge rather than simply setting the gauge to a certain value. The left most point of the gauge stopped the water, while the right most point poured it at the maximum rate. When each task began, the bucket was initially filled with random amount of water and the participants were required to completely fill the bucket while keeping the amount of overflow to minimum. We considered the task to be successful when the bucket was filled more than 95%.

5.4 Measures

Similar to the Study 1, we measured completion time to quantitatively evaluate the general performance and SUS scores [35] and rankings of the participants' preferred interfaces to qualitatively assess the usability and preference of the interface.

5.5 Procedure

The experiment was a within-subjects design with 2 gestures (mid-air touch and wrist rotation), 3 safeguard mechanisms (*ABS*, *AM*, and *AT*), and 3 tasks (*Lamp*, *Speakers*, and *Water faucet*) as independent variables, resulting in 18 different conditions. Completion time, SUS scores and user rankings were measured as dependent variables. In addition, the video feeds displayed on the HMD were recorded for the purpose of qualitative analyses. Latin square design was used for the combination of gestures and safeguard mechani-sms to eliminate ordering effects and the order of the devices was randomized for each task condition.

The participants performed the device configu-ration tasks for 6 different conditions (2 gestures × 3 safeguards). For each condition, they made 30 configurational changes (10 changes for each device). The order of the devices was randomized while keeping each device to appear exactly 10 times. Before each condition, they were informed about the usage and allowed to practice until they felt comfortable. Once the task began, they were asked to configure the devices based on the instruction displayed on the screen. They looked at the instructed device by turning their body and head to trigger the meter to appear and then use the selected gesture and safeguard mechanism to control it. They filled out three separate SUS forms for each of the three devices after each task condition. The completion time was also measured for each task. Finally, we conducted post-condition interview to ask how they felt about the combination of the gesture and the interface mechanism they had just used, right after they filled out the SUS forms for each condition. After all 6 conditions, post-experiment interview was conducted.

Similar to the Study 1, we did not explain the real purpose of each of the three different mechanisms until the end of the experiment to prevent any bias towards a certain interface. During the post-experiment interview, we asked each participant to rank the 6 different combi-nations of gestures and safeguard mechani- sms based on the usability preference. Then, we explained about the safety concerns and the purpose of each safeguard feature designed to advise involuntary inputs and asked each partici-pant to re-rank the combinations based on safety, followed by the overall preference considering both usability and safety. In order to help them recall on their experience, we showed them the recorded video feed. In order to increase the quality of the interview within the limited amount time, we marked all the points where the participants struggled or anything interesting happened during the tasks, and conducted the interview focusing primarily on them.

5.6 Results

5.6.1 Task Completion Time

A three-way repeated measures ANOVA was conducted to examine the effect of gestures, safety features, and task types on completion time. There were statistically significant main effects of gestures ($F_{1,11} = 22.827,\ p < .001$), safety features ($F_{2,22} = 29.274,\ p < .001$), and task types ($F_{1.1,11.8} = 48.741,\ p < .001$) on completion time. Bonferroni post- hoc analysis has revealed that the mid-air touch gesture was slower than the wrist-rotation gesture ($p = .001$), the safeguard mechanisms performed faster in the order of ABS, AT, and AM ($p < .03$). Also, the participants performed faster for the tasks on the devices in the order of *Lamp*, *Speakers*, and *Water faucet* ($p < .05$).

Although the main effects of gestures and safety features were consistent with those of the Study 1, the significant difference between ***Lamp*** and ***Speakers*** was unexpected. While the ***Water faucet*** task was expected to be different due to its task design from the other two, we expected ***Lamp*** and ***Speakers*** tasks to yield the similar result since they were basically identical tasks on different devices. This was largely on account of the devices' location: ***Lamp*** placed right in front of the user was relatively easier to reach, compared to the Speakers on the left requiring the participants to turn their bodies and heads. In addition, as some participants reported the difficulty of controlling the object on the left using their right hands (P1 and P4), we believe such physical restrictions explain why the ***Speaker*** task was slower than the ***Lamp***. There were also significant interactions between the device type and the other factors (gestures and safety features) ($p < .05$).

5.6.2 SUS Scores

The average SUS scores were calculated for each gesture-and-safeguard combination (Fig. 11) and revealed that all three safeguard features scored above average (> 68) with wrist-rotation gesture while only ***ABS*** scored above average with mid-air touch gesture. This was mainly caused by the higher complexity of the experiment where the participants' general performance using the mid-air touch gesture was degraded. However, the order of the safety features based on the usability stayed consistent with the result of Study 1 (***ABS*** > ***AT*** > ***AM***).

5.6.3 Rankings and Interview

At the end of the experiment, we collected user rankings of 6 combinations of the gestures and the safeguard features based on three different aspects: usability, safety, and overall Table 2. In order to compare the difference among the safeguard mechanisms, we conducted Friedman tests for each gesture separately. For mid-air touch, there was a statistically significant difference in terms of usability, $x^2(2) = 10.67$, $p < .005$, and post hoc analysis of Wilcoxon signed-rank test with Bonferroni correction revealed that ***ABS*** was significantly better than ***AM*** ($p < .003$) while there was no significant difference between ***ABS*** and ***AT***. In terms of safety, there was also a significant difference, $x^2(2) = 12.5$, $p < .002$, and post hoc analysis revealed that both ***AT*** and ***AM*** were significantly safer than ***ABS*** ($p = .007$). For the Overall, there was no significant difference. The results were identical for the wrist-rotation gesture with only minor differences, except for the Usability where ***ABS*** was significantly better than ***AT*** in addition to the result of the mid-air touch.

Table 2. Average rankings of the gestures and the safeguard mechanisms from Study 2 (lower number indicates better ranking). Winners of each aspect per gesture are highlighted.

	Mid-air Touch			Wrist Rotation		
	ABS	AM	AT	ABS	AM	AT
Usability	3.33	5.00	4.42	1.33	3.67	3.25
Safety	5.42	3.50	3.17	4.42	2.33	2.17
Overall	4.50	4.67	3.75	2.67	3.25	2.17

During the interview, we found that ***AT*** had a better safety score than ***AM***, as many of the participants reported ***AT*** felt safer than ***AM*** (S2P1, S2P3, S2P5, and S2P6), while it was the opposite in the Study 1. One of the participants even showed his preference of using ***AT*** towards volume control, "I like this one as it would be disturbing if the volume suddenly rises (S2P1)." One of the reasons that ***AM*** received lower safety score was that some participants felt ***AM's*** safety feature of

Table 3. Categorization of SCARII identified in Study 2 including examples and design guidelines.

Category	Examples	Design Guideline
Gestures for selection/ deselection were not clearly distinguishable from changing gestures.	Unintended changes on the meter occurred when users were only trying to select/deselect. Unintended deselection occurred when users were changing values on the meter.	Selection/deselection gestures should be orthogonal from changing gestures.
AR markers were lost, resulting in the failure of device recognition.	As users were focusing on the gauge, their head naturally followed its horizontal movement and thus, the markers went outside of the vision. For *Water faucet*, users only focused on the bucket underneath and forgot to see the faucet above.	The configuration GUI, the target device, and the monitoring device should be placed within the same visible range.
Time pressure and users' general tendency to perform quickly caused human errors.	For *AM*, users overshoot when they tried to snatch the current gauge value. For *AT*, users sometimes selected any random place even before they recognized their hand position.	Safeguard features should be designed not to be affected by users shortcuts or routines which could result in making mistakes.
Unintended locations on the meter were selected, causing accidental changes.	Users selected unintended locations by skill-based mistakes. System sometimes failed to recognize the correct hand position resulting in random changes on the meter.	Safeguard feature should advise involuntary inputs not to cause any abrupt changes. Methods to filter out unstable gesture recognition would be helpful.
Fatigue caused the misusage of gestures.	Due to the arm fatigue, users sometimes failed to keep their hands within the visible range. Some users self-developed less-tiring routines which did not always register to the system correctly.	Avoid repeated use of the gestures that may lead to arm fatigue. If fatigue is inevitable, there should be ways to filter out incorrect use of the selected gestures.
Human errors were made when gesture recognition fails occasionally.	Users randomly waved their hands when the gesture recognition failed. Failure in the gesture detection made some users to create their own routines which only worked temporarily.	Only the reliable gestures should be used for safety-critical situations. Unrelated random hand motion should be ignored by the system.
Other interesting observations.	Physical limitations were present, such as pointing very left with a right hand and turning a wrist extremely to each end. Users confused among the three safeguard mechanisms during the experiment, such as trying to use *AM* like *AT*. For *Water faucet*, users sometimes carelessly turned the water at the maximum rate in order to finish the task quickly.	It is better to place devices towards the center and to utilize middle range of gestures and avoid each end (i.e. making a user to touch left/right ends with right/left hands respectively or to rotate one's wrist to each end). Use consistent gestures and safeguard features in order to reduced mistakes caused by confusion.

snatching the current gauge value to be bothersome and they occasionally made mistakes such as overshooting during the snatching (S2P1, S2P5, S2P7, and S2P11). Overall, many of the results seem to be consistent with the Study 1 and *AT* was fairly successful in providing the balance between usability and safety. Although there were minor complaints on the *AT*' s slow trailing speed (S2P1, S2P8, and S2P9), we believe it can be optimized through additional user study by finding the intersection point between efficiency and safety.

We also asked the participants whether they were able to feel that the safety problems could actually occur based on their experience during the experiment. Many of the participants answered that they actually experienced SCARII issues (S2P6, S2P7, S2P8, and S2P9) especially with

Speakers and *Water faucet*. Also, two participants said that if it were a gas stove, SCARII would be even more critical (S2P9 and S2P10). The reason that the participants did not experience any problems with *Lamp*, was due to the limitation of the video based see-through HMD; even the brightest *Lamp* settings did not generate annoying glare through the display of the Oculus Rift. We believe the result would be different if Optical see-through HMDs are used. On the other hand, it was very interesting to observe that the participants actually felt SCARII issues while they controlled the simulated *Water faucet*. We interpret this as non-photorealistic virtual contents could still be felt as real and affect users' behaviors possibly causing completely new types of safety problems. We believe such direction would be valuable to study in the future.

5.6.4 Video Coding and Analysis

In order to find important design features for developing interfaces used for configuring devices in AR environment, we analyzed the video feeds recorded during the experiment. All the safety related problems considering both the system and human failures [29] were marked through iterative coding [38-39] to ensure the reliability and the resulting inter-coder agreement assessed with Cohen's Kappa [40] showed substantial agreement between the two authors (coders), $x = .748$, $p < .001$. After analyzing problems, we categorized highly frequent safety issues into several different categories (Table 3). Based on the findings, we suggest design guidelines to address the concerns.

6. Discussion

Many of the results from the two studies seem to be consistent with each other indicating that *AT* was fairly successful in providing the balance between both usability and safety as one participant said, "it sits well in the middle (S2P10)." However, during the studies, we found that in some cases, our safeguard features were the causes of the other problems. For example, some users, under time pressure, tried to quickly snatch the current gauge value and mistakenly overshot, causing sudden changes of the gauge value close to the minimum or the maximum. One participant even "did that on purpose to quickly grab the gauge (S2P11)." As users often try to find shortcuts or their own routines, it seems to be important to design the safeguard features which are not affected by such user behaviors.

One of the biggest problems of the mid-air touch gesture was that the ***selection*** and the ***deselection*** were not orthogonal to the ***change***. A number of times, the gauge value was affected and ***changed*** when the participants just tried to ***deselect*** because ergonomically it was nearly impossible for them to precisely pull back their hands to ***deselect*** without any horizontal movement which affected the gauge value. One participant was very annoyed by this and created her own routine which worked only for a part of times, exacerbating the situation after being confused by her own routine. Agitated by the experience, she said, "I feel it is deliberately designed to make people upset (S2P12)." On the other hand, for all participants including S2P12, such issues did not occur while using the wrist-rotation gesture where the changing action was orthogonal to the selection and deselection. The result clearly emphasizes that it is important to avoid using any combination of actions which affect each other.

Another common issue was that the participants often lost track of the AR markers by moving their heads without noticing. As they were focusing on the gauge of the meter, their heads naturally followed its horizontal movement and, as a result, the marker went outside of the vision causing the meter to disappear even though the task had not been finished. In addition, for the *Water faucet* task, as device being monitored during the task (the bucket) was apart from the device being controlled (the faucet), some participants reported that "it was difficult to have both of them in a single eyesight (S2P4)." In order to alleviate such problems, we suggest placing the configuration GUI, the target device and the device being monitored within the same visible range.

Fatigue was also a critical reason for causing SCARII as it led to misusing gestures. Towards the end of the experiment, some participants had difficulties keeping their hands within the visible range (S2P2 and S2P11). There were even participants who self-developed less-tiring routines, such as dropping their hands for deselection instead of pulling them back (S2P3 and S2P10), which did not always register to the system correctly. Although each participant had to perform many tasks using same gestures multiple times under the experimental settings, in real scenarios, repeated use of the gestures that may lead to arm fatigue should be avoided. In addition, if such redundant usages which cause arm fatigue are inevitable, the system should be able to filter out incorrect use of the selected gestures.

One of the less frequent but highly interesting observation was identifying the presence of physical limitations. Some participants seemed uncomfortable touching the very left end of the meter with their right hands (S2P2, S2P3, S2P4, S2P9, S2P11, and S2P12) or rotating their wrist to each end (S2P8, S2P9, S2P10, and S2P11). "Targets on each corner was difficult to reach… small range in the middle was just fine (S2P9)." In other words, it is important to utilize a small range from the center and avoid each end, keeping users from extremely stretching their arms or exceedingly twisting their wrists. Moreover, big gestures which require users to wave their arms would not be "socially acceptable in public places (S2P4)."

Although it was not classified as an error, one participant showed an interesting behavior where he snatched the current gauge value using ***AT*** only when the volume of the ***Speakers*** was very high. During the interview, he explained, "the music was so loud that I wanted to quickly reduce the volume (S2P4)." While our safeguard features focused primarily on preventing SCARII from happening, we discovered that it could be as important to design an interface which helps users easily recover when inevitable safety problems actually occur. In such recovery scenario, using an interface with the greatest efficiency, such as ***ABS***, would be the safest choice as it minimizes the user's exposure to the dangerous condition. Providing a simple and quick mode switching method among safeguard features would be one possible approach to support both the prevention of and the recovery from SCARII.

There was a participant who thought mid-air gestures "were not ready enough to be used in safety-critical situations (S2P3)" leaving us some space for the future study. As people become more conservative when the interface involves the safety concerns, such reluctance can also become a non-system-related barrier that needs to be overcome. On the contrary, one participant acted cautiously only for the device he felt unsafe; the *Water faucet*. He always started from the minimum position and increased the amount very slowly. "It felt natural to carefully start from the slow rate (S2P10)." Therefore, well-designed visual affordances on the objects with safety-concerns can complement imperfect mid-air gesture recognitions. We believe this would also be a promising direction for the future study.

7. Conclusion

In order to address potential SCARII, we designed an interface with four different safeguard mechanisms. We applied the interface to two different gestures, mid-air touch and wrist-rotation, and evaluated its usability and safety. We found that *Absolute-Trail (AT)* was appreciated by its balance in safety and usability. In addition, we conducted an observational study to explore various SCARII issues that could rise in real life. Through the qualitative analysis, we categorized discovered SCARII issues and developed design guidelines for avoiding them.

Acknowledgement

This work was supported by Samsung Electronics.

References

〔1〕 Z. Huang, W. Li, P. Hui, "Ubii: Towards Seamless Interaction between Digital and Physical Worlds," In Proceedings of the 23rd ACM international conference on Multimedia (MM '15), pp.341-350, 2015.

〔2〕 S. G. Sakamoto, L. C. de Miranda, H. Hornung, "Home control via mobile devices: State of the art and hci challenges under the perspective of diversity," In Universal Access in Human-Computer Interaction. Aging and Assistive Environments, pp.501-512, 2014.

〔3〕 Y. Suo, C. Wu, Y. Qin, C. Yu, Y. Zhong, Y. Shi, "HouseGenie: Universal Monitor and Controller of Networked Devices on Touchscreen Phone in Smart Home," In Proceedings of the 2010 Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing (UIC-ATC '10), pp. 487-489, 2010.

〔4〕 T. Lee, M. Moon, "Design and Development of Mobile Air-Quality Measurement System based on IoT Platform," The Journal of Korean Institute of Next Generation Computing, Vol. 12, No. 3, pp. 75-84, 2016.

〔5〕 M. Moon, "IoT based Garbage Collection Management System Through Volume Prediction," The Journal of Korean Institute of Next Generation Computing, Vol. 13, No. 1, pp. 45-53, 2017.

〔6〕 J. Matero, A. Colley, "Identifying unintentional touches on handheld touch screen devices," In Proceedings of the Designing Interactive Systems Conference (DIS '12), pp. 506-509, 2012.

〔7〕 W. Hürst, C. V. Wezel. "Gesture-based interaction via finger tracking for mobile augmented reality," Multimedia Tools and Applications, Vol. 62, No. 1, pp. 233-258, 2013.

〔8〕 J. P. Wachs, M. Kölsch, H. Stern, Y. Edan, "Vision-based hand-gesture applications," Communications of the ACM, Vol. 54, No. 2, pp. 60-71, 2011.

〔9〕 P. Belimpasakis, R. Walsh, "A combined mixed reality and networked home approach to improving user interaction with consumer electronics," Consumer Electronics, IEEE Transactions on 57, No. 1, pp.139-144, 2011.

〔10〕 A. A. de Freitas, et al., 2016. "Snap-To-It: A User-Inspired Platform for Opportunistic Device Interactions," In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16), pp. 5909-5920, 2016.
〔11〕 J. Lee, et al., "A unified remote console based on augmented reality in a home network environment," Consumer Electronics, 2007 Digest of Technical Papers International Conference on Consumer Electronics, 2007.
〔12〕 S. Mayer, N. Inhelder, R. Verborgh, R. V. de Wallet, "User-friendly configuration of smart environments," In Pervasive Computing and Communications Workshops (PERCOM Workshops), pp. 163-165, 2014.
〔13〕 S. Mayer, M. Schalch, M. George, G. Sörös, "Device recognition for intuitive interaction with the web of things," In Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication, pp. 239-242, 2013.
〔14〕 S. Mayer, G. Soros, "User Interface Beaming—Seamless Interaction with Smart Things Using Personal Wearable Computers," In Wearable and Implantable Body Sensor Networks Workshops (BSN Workshops), 2014 11th International Conference on, pp. 46-49, 2014.
〔15〕 T. Piumsomboon, A. Clark, M. Billinghurst, A. Cockburn, "User-defined gestures for augmented reality," In Human-Computer Interaction - INTERACT 2013, pp. 282-299, 2013.
〔16〕 H. Pohl, M. Rohs, "Around-device devices: my coffee mug is a volume dial," InProceedings of the 16th international conference on Human-computer interaction with mobile devices & services (MobileHCI '14), pp. 81-90, 2014.
〔17〕 H. Benko, E. W. Ishak, S. Feiner, "Cross-dimensional gestural interaction techniques for hybrid immersive environments," IEEE Proceedings. VR 2005. Virtual Reality, 2005., Bonn, pp. 209-216, 2005.
〔18〕 J. D. Hincapié-Ramos, K. Ozacar, P. P. Irani, Y. Kitamura, "GyroWand: IMU-based Raycasting for Augmented Reality Head-Mounted Displays," In Proceedings of the 3rd ACM Symposium on Spatial User Interaction (SUI '15), pp.89-98, 2015.
〔19〕 C. Corsten, I. Avellino, M. Möllers, J. Borchers, "Instant user interfaces: repurposing everyday objects as input devices," In Proceedings of the 2013 ACM international conference on Interactive tabletops and surfaces (ITS '13), pp. 71-80, 2013.
〔20〕 V. Heun, S. Kasahara, P. Maes, "Smarter objects: using AR technology to program physical objects and their interactions," In CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13), pp.961-966, 2013.
〔21〕 V. Heun, J. Hobin, P. Maes, "Reality editor: programming smarter objects," In Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication (UbiComp '13 Adjunct), pp.307-310, 2013.
〔22〕 H. Kim, K. Jeong, S. Kim, T. Han, "Ambient Wall: Smart Wall Display interface which can be controlled by simple gesture for smart home," In SIGGRAPH Asia 2011 Sketches (SA '11), pp. 1-2, 2011
〔23〕 C. Kühnel, et al., "I'm home: Defining and evaluating a gesture set for smart-home control," International Journal of Human-

Computer Studies, Vol. 69, No. 11, pp. 693-704, 2011.

[24] D. Lee, et al., "Actual remote control: a universal remote control using hand motions on a virtual menu," IEEE Transactions on Consumer Electronics, Vol. 55, No. 3, pp. 1439-1446, 2009.

[25] K. Tsukada, M. Yasumura, "Ubi-finger: Gesture input device for mobile use," Ubicomp 2001 Informal Companion Proceedings, Vol. 11, 2001.

[26] J. Schwarz, R. Xiao, J. Mankoff, S. E. Hudson, C. Harrisonm "Probabilistic palm rejection using spatiotemporal touch features and iterative classification," In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14), pp. 2009-2012, 2014.

[27] F. Kerber, P. Schardt, M. Löchtefeld, 2015. "WristRotate: a personalized motion gesture delimiter for wrist-worn devices," In Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia (MUM '15), pp. 218-222, 2015.

[28] M. Na, H. You, T. Kim, "A Vision-based Real-time Hand Pose and Gesture Recognition Method for Smart Device Control," The Journal of Korean Institute of Next Generation Computing, Vol. 8, No. 4, pp. 27-34, 2012.

[29] D. Embrey, "Understanding human behaviour and error," Human Reliability Associates 1, pp. 1-10, 2005.

[30] T. Masui, T, K. Kashiwagi, G. R. Borden IV, Elastic graphical interfaces to precise data manipulation, In Conference companion on Human factors in computing systems, pp. 143-144, 1995.

[31] H. J. Chae, et al., "Prevention of Unintentional Input While Using Wrist Rotation for Device Configuration," In Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16), pp. 205-206, 2016.

[32] A. T. Cabreira, F. Hwang, "An analysis of mid-air gestures used across three platforms," In Proceedings of the 2015 British HCI Conference (British HCI '15), pp.257-258, 2015.

[33] M. Rahman, S. Gustafson, P. Irani, S. Subramanian, "Tilt techniques: investigating the dexterity of wrist-based input," In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09), pp. 1943-1952, 2009. DOI=http://dx.doi.org/10.1145/1518701.1518997

[34] G. D. Langolf, D. B. Chaffin, J. A. Foulke, "An investigation of Fitts' law using a wide range of movement amplitudes," Journal of Motor Behavior 8, pp.113-128, 1976.

[35] J. Brooke, "SUS-A quick and dirty usability scale," Usability evaluation in industry, Vol. 189, No. 194, pp. 4-7, 1996.

[36] E. R. Girden, ANOVA: Repeated measures, Sage, No. 84, 1992.

[37] J. Brooke, "SUS: A Retrospective," Journal of Usability Studies, Vol. 8, No. 2, pp.29-40, 2013.

[38] V. Braun and V. Clarke, "Using thematic analysis in psychology," Qualitative Research in Psychology, Vol 3, No. 2, pp. 77 - 101, 2006.

[39] D. J. Hruschka, et al., "Reliability in Coding Open-Ended Data: Lessons Learned from

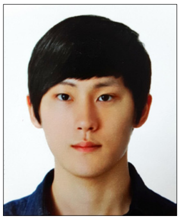
HIV Behavioral Research," Field Methods Vol. 16, No.3, pp. 307-331, 2004.
〔40〕 J. Cohen, "A Coefficient of Agreement for Nominal Scales," Educational and Psychological Measurement, Vol. 20, No. 1, pp. 37-46, 1960.

저자소개

◆ **채한주**

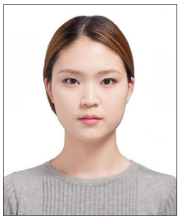
- 2009년 카네기멜론대학교 전기.컴퓨터공학 학사
- 2010년 카네기멜론대학교 전기.컴퓨터공학 석사
- 2015~현재 서울대학교 컴퓨터공학부 박사과정
- 관심분야: 증강현실, 인간-컴퓨터 상호작용

◆ **황정인**



- 2015년 서울대학교 컴퓨터공학 학사
- 2017년 서울대학교 컴퓨터공학 석사
- 2018년~현재 NCSOFT 연구원
- 관심분야: 증강현실, 인간-컴퓨터 상호작용, 자연어처리

◆ **김이은**



- 2012년 서울대학교 컴퓨터공학 학사
- 2017년 서울대학교 컴퓨터공학 석사
- 2019년~현재 삼성리서치 연구원
- 관심분야: 인간-컴퓨터 상호작용

◆ **고 건**

- 2009년 일리노이 주립대 어바나 샴페인 컴퓨터과학 학사
- 2017년 서울대학교 컴퓨터공학 박사
- 2017년~현재 제우스리빙 소프트웨어 엔지니어

◆ **서진욱**

- 1995년 서울대학교 계산통계학과 전산과학전공 학사
- 1997년 서울대학교 컴퓨터공학 석사
- 2005년 메릴랜드 대학교 컴퓨터공학 박사
- 2005년~2006년 Children's Research Institute 연구원
- 2006년~2008년 Children's Research Institute & George Wachington University School of Medicine 조교수
- 2009년~현재 서울대학교 컴퓨터공학부 전임교수
- 관심분야: 인간-컴퓨터 상호작용, 정보시각화