GestureDrawer: One-Handed Interaction Technique for Spatial User-Defined Imaginary Interfaces

Teo Babic
BMW Group
Munich, Germany
teo.babic@bmw.de

Harald Reiterer
HCI Group
University of Konstanz
Konstanz, Germany
harald.reiterer@uni-konstanz.de

Michael Haller
Media Interaction Lab
University of Applied Sciences Upper Austria
Hagenberg, Austria
michael.haller@fh-hagennberg.at

ABSTRACT

Existing empty-handed mid-air interaction techniques for system control are typically limited to a confined gesture set or point-and-select on graphical user interfaces. In this paper, we introduce *GestureDrawer*, a one-handed interaction with a 3D imaginary interface. Our approach allows users to self-define an imaginary interface, acquire visuospatial memory of the position of its controls in empty space and enables them to select or manipulate those controls by moving their hand in all three dimensions. We evaluate our approach with three user studies and demonstrate that users can indeed position imaginary controls in 3D empty space and select them with an accuracy of 93% without receiving any feedback and without fixed landmarks (e.g. second hand). Further, we show that imaginary interaction is generally faster than mid-air interaction with graphical user interfaces, and that users can retrieve the position of their imaginary controls even after a proprioception disturbance. We condense our findings into several design recommendations and present automotive applications.

CCS CONCEPTS

• **Human-centered computing~Gestural input** • **Human-centered computing~User interface design** • **Human-centered computing~User centered design**

KEYWORDS

Interaction technique: gestures; imaginary interfaces; spatial user interfaces; screen-less; ubiquitous computing

ACM Reference format:

T. Babic, H. Reiterer, M. Haller. 2017. GestureDrawer: One-Handed Interaction Technique for Spatial User-Defined Imaginary Interfaces. In *Proceedings of the 2017 Symposium on Spatial User Interaction.* ACM, Brighton, United Kingdom, October 2017 (SUI '17).
DOI: 10.1145/3131277.3132185

1 INTRODUCTION

With the improvements in computer vision and tracking systems [33], we are able to introduce hand-gesture input, where users no longer have to hold, wear, or retrieve any physical input device. Instead, users can use an imaginary interface (i.e. invisible controls positioned in 3D space) to interact with the system while observing the effect of their gestures from any output in the environment - no need to visually focus on the graphical user interface (GUI) displayed on a physical screen to enable interaction. There are many use cases where such interfaces can be beneficial. Our motivation to explore imaginary interfaces was driven by BMW, an automotive company, looking for interaction concepts for a level 3 autonomous driving system, where the driver does not need to drive for a certain period of time. Ideally, in such scenarios, the in-car interaction remains empty-handed and eyes-free, for the case that the driver needs to retake control on short notice (4 s). Simultaneously, it should offer more expressiveness then the status-quo in-car input devices designed for low distraction (e.g. rotary knobs, simple touchscreen inter-faces). Mid-air interaction found its way in the interiors of some car manufactures [5]. Nevertheless, these systems are still very limited, where only a few shortcuts can be triggered by semaphoric gestures. Following this idea, we asked ourselves to what extent a user (driver) can interact with one free hand in mid-air by using an imaginary interface (completely eyes-free).

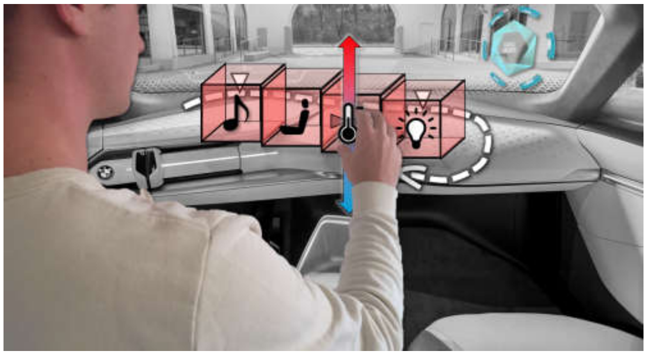

Figure 1: Automotive use-case scenario for *GestureDrawer*.

The Imaginary Interfaces demonstrated a two-handed approach for screen-less interaction, where users were holding an invisible canvas with one hand and used the second hand to interact on it, e.g. point at targets, input single-stroke letters, create and annotate simple drawings [12]. In similar papers [11, 13, 14, 20, 23, 24, 34], users also had to use both hands to interact, which increased fatigue, compared to a one-handed approach and made it impractical for many potential use cases, e.g. where users only have one hand available and/or cannot visually

---

SUI '17, October 16–17, 2017, Brighton, United Kingdom

ACM ISBN 978-1-4503-5486-8/17/10…$15.00
https://doi.org/10.1145/3131277.3132185

attend to a visible land-mark or user interface. Additionally, all these related projects focused on interactions happening on a 2D imaginary canvas placed in a 3D space. Thus, the full expressiveness of 3D gestural imaginary interaction remained unused.

2 GESTURE DRAWER

In this paper, we introduce *GestureDrawer*, a novel one-handed interaction technique, consisting out of two actions; we open a 3D horizontal menu (further called *drawer*) and manipulate its controls in up to three *Degrees of Freedom* (DOF*)* by moving our hand though 3D space. The interplay between these two actions results in the following interaction principle, see Figure 2:

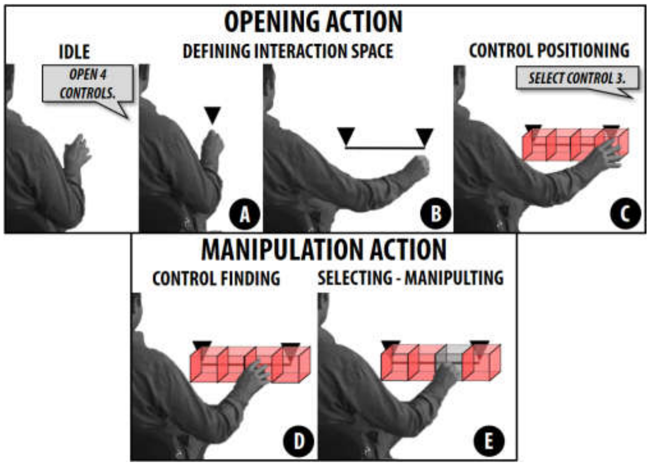

Figure 2: Actions of the *GestureDrawer* interaction technique: Opening action defines a gestural interaction space and fills it with imaginary controls. Manipulation action enables selection or manipulation of those controls in up to 3DOF.

2.1 Opening Action

Firstly, we define the starting point (anchor) of the *drawer* by making a grasp gesture on a preferred position in 3D space, see Figure 2a. Once the anchor is set, we hold-on to the grasp gesture and open the *drawer* with a horizontal hand-motion to the right, see Figure 2b. Next, the grasp gesture is released, setting the end-point and the length of the *drawer* based on the length of the hand motion. A certain number of interface controls can then be positioned into the *drawer*. In Figure 2c, for example, four controls are positioned into the *drawer*. If we would open a smaller or larger *drawer* (by making a shorter or longer gesture, respectively), the size of each control would scale accordingly along all three Cartesian axes. Simultaneously, by performing the *drawer* opening action, we acquired an understanding of the interaction space (i.e. anchor, length and end-point of the *drawer)* via proprioception and visuospatial means. With this knowledge, we then create an imaginary user interface, by logically dividing the *drawer* into multiple areas (controls) and store it in our visuospatial short-term memory. Our imaginary interface then enables us to imagine the position of each control in the real-world 3D space.

2.2 Manipulation Action

Afterwards, once we want to select or manipulate a certain control, we retrieve the control's position from our imaginary interface and move our hand to the appropriate position in empty 3D space; in Figure 2d, for example, it would be control three. Just by using imagination and without any visual feedback, we simply grasp and immediately release a single control to make a selection, see Figure 2e. If we hold on to the grasp gesture and move our hand, we can manipulate the control's functionality in one to up to three DOF (in Figure 1, for example, the driver can adjust the temperature by moving his hand upwards or downwards).

2.3 Benefits and Contribution

First, we present the design of the *GestureDrawer* interaction technique, whose contribution can be described as follows:

- *3D imaginary interface*: It creates a full user interface (structure, number, position and the scale of its controls) in 3D and communicates it to the user´s short-term memory for immediate interaction, without needing compulsory learning phases to learn the positions of the controls and without needing visual feedback (instant eyes-free interaction).
- *3D interaction*: It is the first imaginary interface allowing the arbitrary positioning of interface controls in the 3D space, as well as selection and 3D manipulation of those controls, going beyond ray-casting or 2D plane based input techniques.
- *One-handed*: It is the first one-handed interaction technique for imaginary interfaces and the first which does not use fixed landmarks (e.g. objects from the environment, additional body parts) and therefore reduces hand fatigue compared to two-handed approaches.
- *User-definition*: Users can self-define the position and scale of the interface controls (interaction space) for the benefit of comfort (users can define the interaction space in good reach and comfortable position), interaction precision (possibility to scale controls) and mobility.

Second, we present three user studies, which were conducted for finding benefits and exposing limitations of our design:

- *Accuracy*: *GestureDrawer* allows fast and accurate user interface access, where users can select up to four interface controls with an accuracy of 93% in empty-space, without any form of user interface representation (e.g. visual, auditory, tactile). Additionally, it allows 3D manipulation of its controls, with a fairly good (70%) *drawer* position revisitation accuracy; once the users hand is multidimensionally aside after a 3D manipulation task.
- *Speed*: It promotes the usage of user's visuospatial memory resulting in a cognitively light-weight interaction concept which does not rely on visual-search and is therefore also faster than the state-of-the-art point-and-select gesture controlled GUI.

Finally, we propose implications for design and present *GestureDrawer* applications.

3 RELATED WORK

Our work builds on related work on imaginary interfaces, hand-gesture input, cognitive psychology and spatial interaction. With this work we avoided the state-of-the-art approach which in many cases does not achieve success as discussed in [4, 6, 36]; designing first a complex user interface and then trying to find simple, easy-to-use gestures for it.

3.1 Removal of the Graphical User Interface

The strong dependency of GUIs on physical displays has three considerable disadvantages, which encouraged us to investigate imaginary interfaces: (1) it limits user's freedom and flexibility (in contrast, a user can ideally move, sit, stand and look wherever he would like to, while performing hand gestures). Subsequently, he could observe effects of his actions from various output modalities, a perfect example is the in-car interaction (haptic – seat massage, auditory – volume, vestibular sense – seat adjustment or driving mode change by acceleration, thermoception – air condition intensity). (2) Visually attending to a GUI is not only cumbersome, but also consumes cognitive capacities that could otherwise be directed to the task at hand [6, 30]. (3) Once the user acquired spatial knowledge about a user interface, this frees users' cognitive resources, decreases frustration, preserves time otherwise needed for visual search and reduces the chance of encountering a gulf of execution [28, 30].

3.2 Visuospatial Short-Term Memory

Baddeley and Hitch's model of working memory includes a subcomponent called the *visuospatial sketchpad* which is used for temporary storage, maintenance, and manipulation of both visual and spatial information [2]. Proprioception is the human sense of the relative location of the parts of the human body and is a key component in working memory (short-term) and muscle memory (long-term) [9, 31]. To avoid complicated physiological models of memory and to stay consistent with related work on imaginary interfaces [11, 12, 13, 14, 30] we will refer throughout this paper to people's short term memory capacity to remember and recall locations in space as *visuospatial memory*.

In general, it is known that short-term visuospatial memory is limited with information fading over short periods of time [30]. Participants in Gustafson et al.'s study [12] demonstrated reduced accuracy in drawing long, complex gestures, suggesting that participants' visuospatial memory for the locations of earlier strokes was fading over time. However, attempts to measure the duration of short-term spatial memory have had varied results and strongly dependent on how the memory is being acquired and which exact part of the memory is being allocated [2]. Nevertheless, it is clear that people are able to retain visuospatial information for a limited time.

3.3 Spatial Reference Systems

To locate a user interface controls in empty space, researchers proposed the usage of the non-dominant hand as a proprioceptive frame of reference or landmark (e.g. L-shaped gesture holding the imaginary canvas [12]). The non-dominant hand helped the interacting hand to locate certain positions in relation to it. Like frames of reference, the information provided by landmarks is usually not as rich – a landmark normally serves to identify a single point in space, rather than imply a set of axes through it. Using the non-dominant hand to create a proprioceptive frame of reference increased users' accuracy in a memory task [16] or pointing task [12] and alleviated effects caused by disorientation (such as physically turning the body around between tasks) [12]. Interestingly, it was also shown that the task completion time improved when the landmark is not present, since the alignment of the two hands caused additional cognitive workload [12]. Some projects used the non-dominant hand in a different manner, they used the user's forearm [24] or palm [13, 14], as a frame of reference and simultaneously as a touch interface with invisible controls. It was show that in an eyes-free manner, users could divide their forearm, limited by the wrist and elbow as landmarks, into 5 regions and select each of them with an accuracy of 95%. To remember a certain control position in empty space, without the usage of the second hand or short-term memory, some works used extensive learning phases [8, 10, 37]. In this learning phases, they guided the users hand with the help of visual guidance (displayed on a mobile device [37] or big screen [8, 10]) to certain positions in empty space. After users revisited each position over and over again, they gained enough muscle memory (long-term memory), eventually allowing eyes-free interaction. The downside of this approach are that the users cannot interact instantly in an eyes-free manner, they need to do explicit learning prior to the actual interaction, and that the setup requires a physical device for the guidance visualization.

3.4 Input by Hand Gestures

Many researchers proposed free-handed gesture input recognition systems, for example, ubiquitous computing [1], wearables [12, 35], phones [18], head-mounted displays [32], cars [5] and on-body interaction [24]. To control these interfaces various gestural interaction techniques have been investigated (e.g. semaphoric, pointing, direct manipulation) [1]. The field of natural user interfaces addresses various problems produced by hand gesture input [4, 27, 36] (e.g. hand fatigue [15, 29], reach [22, 25], public acceptance [17], learning gesture sets [26], Midas problem [22, 36]) and they mainly propose that for natural interaction, we need new user interfaces and interface design approaches where user's skill, learning ability and time investment present the core components. These components are especially important for post-WIMP interfaces, where it's not necessary that a good recognition system, gesture set, visualization, precise virtual or augmented immersion, individually will bring products success. Instead, the whole "package" needs to be thoughtfully designed [6, 36].

4 INTERACTION TECHNIQUE

Our main motivation was to go beyond related work and support an empty-handed (no in-hand devices), one-handed (no landmarking by the second hand) and instant eyes-free (no compulsory learning phases) interaction technique, by which users could easily access user interface controls positioned in 3D space and accurately interact with those controls by moving their hand in up to three DOF. All these points led us to design the *GestureDrawer* interaction technique, centered on three core components: *imaginary interface*, *one-handed imaginary interaction* and *user-definition*.

4.1 Imaginary Interface

Instead of a GUI, we propose an imaginary interface, which does not require an observable representation (e.g. visual - display, haptic – physical buttons). It uses a spatial layout of interface controls that is temporarily stored in user's visuospatial short-term memory [2]. By temporarily storing the interface in our memory, we avoid the need for users to extensively learn spatial positions of the interface controls. This is beneficial since skill acquisition is expensive and can create a gulf of competence. Furthermore, it reduces the risk that certain users will not invest adequate time in learning a new interface layout and will never reach a certain skill level [36]. With the *GestureDrawer*, we only have to open the *drawer*, temporarily acquire knowledge about its layout (position of its controls) and once done with the interaction, we can forget about the acquired layout knowledge. Later we can reopen the *drawer* and acquire a fresh imaginary "frame" about its new layout. In-between, our context can even change. For example, a person initially sitting in the first-row seats of a car transitions to sitting in the back, or a person wearing smart-glasses transitions from being in the office to being in a crowd on the subway. In both cases, we could always reopen the *drawer* at a different position relative to our body, while keeping the same functions accessible and the interaction technique consistent.

4.2 One-handed Imaginary Interaction

In order to design a one-handed interaction technique for an imaginary interface, we first need to remove one hand out of the two-handed "interaction equation" proposed by the related work [11, 13, 14, 16, 20, 23, 24, 34]. In other words, we need to remove the most essential part of imaginary interaction – the user's spatial reference frame or landmark (e.g. L-shaped gesture holding the imaginary canvas [12]), which is used for mapping the imaginary interface to the real-world and orienting the interacting hand in empty space. We propose to use the opening action gesture as a retrospective gesture to define a reference frame (i.e. information about a gesture that was done in the past) for the *GestureDrawer* interaction space.

4.3 User-definition

With the opening action, we not only acquire knowledge about the interface, but we also influence its creation (position, scale of its controls). By inducting users' self-definition aspect into the creation process of the user interface, we give users more expressive power, which has a direct influence on many aspects of the interaction. For example, in the case of being a novice user, we can open a larger *drawer* to have larger controls (easier selection) or if we find long-motion gestures socially inappropriate or uncomfortable, we can open a *drawer* on a microgesture level. Subsequently, we do not need to search for the user interface positioned in 3D space, e.g. when we try to find the depth at which the developer placed the 2D interaction plane, which can always be a bit too far or too close for certain users [36]. Furthermore, we do not need to accurately focus our fingers to a certain control's position in empty space (causing jittering), where the control's size can always be too small for certain users. Both cases are well known problems of status-quo gestural interfaces.

4.4 Limitation

The main limitation of our *GestureDrawer* interaction technique is that the user and the system must know how many interface controls can be placed in the *drawer*, and what their respective functionalities are. In practice, this translates to needing one or a combination of the following: a limited number of predefined controls and their functions that later need to be learned by users, a possibility for users to self-define their favorite functions (e.g. most used), or minimal visual hints of functions, that can be retrieved by the *GestureDrawer* in a certain scenario (either on a small screen or without spending big amounts of screen real-estate in general). Our implemented examples can be seen in the application section. Knowledge transfer of control functionalities was investigated in Imaginary Phone [13], where users were able to memorize 64% of the app positions on their iPhone home-screen and recall them later for imaginary interaction. Therefore, we do not cover the human ability to remember associations between functions and spatial positions. Instead, we focus solely on the core components of the *GestureDrawer* and investigate (1) whether users can imagine placing a certain number of controls in 3D empty space without fixed landmarks at all, and (2) if so, how accurately can they retrieve a certain control's position and select it, but also can they manipulate it in up to three DOF and still find the way back to the *drawer* position once the hand moved aside.

5 IMPLEMENTATION

Gesture states (*registration*, *continuation*, *termination*) were generated by using a "grasp" clutching-gesture as this has been reported to be the preferred gesture by many users [19]. Additionally, we added the hand movement to it [3, 7]. We divided the *GestureDrawer* interaction technique into two single-stroke gesture actions to reduce fatigue, by keeping both the physical clutching and hand movement at a minimum.

By limiting the horizontal hand movement to transition from left-to-right (i.e. we cannot open the *drawer* from a top-to-bottom or right-to-left motion), and incorporating two clutch events in the opening action, we ensure that the gesture is sophisticated enough for the sensory system to interpret our interaction intention with minimal false positives. Once the *drawer* is open, we can perform multiple manipulation actions on its controls. With this interplay between the opening and manipulation action, the recognition system either waits for the complete opening action gesture to occur, or in the case that the *drawer* is already open, it only expects clutching (grasp) gestures on specific areas of the *drawer*'s controls, see Figure 3, *left*. *GestureDrawer* can also be opened by a left-handed user by the same opening action (left-to-right). The horizontal direction of the opening action was mainly inspired by Kulshreshth's [21] comprehensive study on graphical menus for mid-air gestures, where the horizontal menu layout was the fastest, most accurate and most preferred by their users for a clutching-based interaction technique.

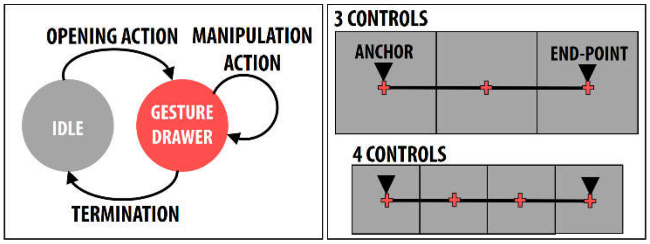

Figure 3: Interchange of the opening and manipulation action (left), control size definition and splitting process (right).

When an opening action is performed, the control size is determined by dividing the *drawer* length by *N*-1, where *N* would be the total number of controls. The center points of the first and last control (*N*) are positioned on the anchor and end-point positions of the *drawer* respectively (Figure 3, *right*). In our implementation, each control was implemented as a 3D cube, so that the selection error tolerance was the same on all axes (*x*, *y* and *z*). As no margin was placed in-between controls, they were positioned immediately adjacent to each other.

6 USER STUDIES

Three empirical user studies were conducted to investigate: (1) how users define the *drawer* and how accurately they can then select its controls, (2) how users' memory retention and performance compare when no GUI is present versus when it is available, and (3) users' ability to revisit the location of a control, after they complete a 3D placement task (translation only); moving their hand away from the *drawer*. The user studies were conducted one after the other with the same subjects.

6.1 Apparatus

The apparatus was the same for all three studies. Participants were seated on an office chair without armrests (so that their right hands could move freely), and were facing an empty wall. For the conditions with visual feedback, a mobile Sharp 70" display was positioned in-front of them. For gesture tracking, we used an off-the-shelf Leap Motion sensor. *GestureDrawer* was implemented in C# and ran on a Windows 10 computer. The Leap Motion sensor was attached to the right side of the apparatus chair, see Figure 4, *left*. Speech synthesis software running on the apparatus computer was used to verbally relay instructions to participants.

6.2 Participants

18 unpaid volunteers (4 female), 25–58 years old ($\bar{x} = 34$, $SD = 8.8$) were recruited from a partner company and a local university. All participants were right-handed. We balanced the duration of each study and their conditions so that a participant was never constantly moving his hand for longer than 8 minutes. That way, a negative influence of hand fatigue on the study results was reduced to a minimum as advised in [15].

6.3 Procedure

At the beginning, we verbally explained the *GestureDrawer* interaction technique. This was done to prevent participants from being influenced by us performing the *GestureDrawer* (e.g. mimicking our anchor position and opening length). We gave participants enough time ($\bar{x} = 6$ min) to get comfortable with the experimental setup. In a dedicated application, they could "play" with the *GestureDrawer* interaction technique. They could open a *drawer* and select its controls, for which they received a sound clip to indicate if the selection was correct or incorrect. It was advised that they should not exaggerate with the size of the drawers to attain higher selection accuracy. Instead they should define a *drawer* size which is comfortable to use, independent of the task's complexity (number of controls). The official study session (including all three user studies) began once participants confirmed that they understood what was required of them.

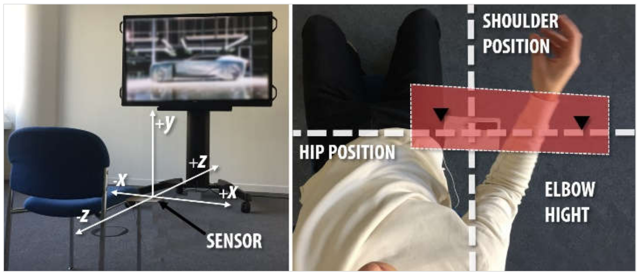

Figure 4: Apparatus (left), average interaction space where users performed *GestureDrawer*, marked in red (right).

6.4 Data Processing and Analysis

We used the Cartesian coordinate system originating from the middle of the Leap Motion sensor to determine the position of participant's hand, see Figure 4, *left*. For statistical analyses, we used repeated measures one-way ANOVA ($\alpha = .05$), where in study 2 we used a two-way comparison. The Greenhouse-Geisser correction was used if the assumption of sphericity was violated. Post-hoc analyses on the main effects were conducted. For all post-hoc pairwise tests we used Bonferroni correction confidence intervals.

7 STUDY 1: DEFINITION AND ACCURACY

The purpose of the first study was to find out where and at what scale participants define and perform the *GestureDrawer* interaction technique. More precisely, we investigated how fine-grained and accurate they can select a control from their self-defined *drawer* (interaction space), and how the size of the *drawer* influences the control selection accuracy.

7.1 Hypotheses

This study was designed and conducted with the following hypotheses in mind:

- *H1.1*: When more controls need to be positioned into the *drawer*, participants will also define larger drawers.
- *H1.2*: The selection accuracy decreases when more controls are positioned within the *drawer*.

7.2 Experiment Design

Five conditions were examined based on the independent variable, LEVEL (complexity), which was ranging from 2 to 6. The LEVEL corresponds with the number of controls that are positioned within the *drawer*. We excluded LEVEL 1, since it would have been too simple, just to place/select only one control. The maximal complexity has been chosen in reference to [24].

Study 1 was conducted using a two-step procedure. Firstly, a LEVEL was specified, after which the participant was expected to perform the *GestureDrawer* opening action in order to position the instructed number of controls in the 3D space. In the second step, the control-to-select was instructed, and participants had to find it and select it, see Figure 2. If the physical position of the participant's selection was outside of the control (implemented as cubic bounding box) the selection was considered incorrect. If the selection point was inside the control, the selection was deemed correct. After each selection, a sound clip was played signifying whether the selection was correct or incorrect – ending one trial. The *drawer* needed to be reopened for each selection. Each control was selected three times per LEVEL condition. Thus, each participant completed a total of 60 trials (i.e. for all 5 LEVEL conditions). The instruction order of LEVEL conditions was counterbalanced across all participants and the control-to-select instruction was randomized.

7.3 Results

Overall, we collected 1,080 control selection and *drawer* opening data entries. Participants completed this study on average in 6 minutes ($SD = 56$ s).

Figure 4, *right*, shows the average interaction space defined by participants while using the *GestureDrawer* interaction technique. We found a significant difference between the lengths of the opening action and the LEVEL conditions ($F_{4,68} = 20.528$, $p = .000$), see Figure 5. The post-hoc test showed a significant difference between the pairs LEVEL 2 and LEVEL 3 ($p = .009$), LEVEL 3 and LEVEL 4 to 6 ($p = .004$) and also between LEVEL 2 and LEVEL 4 to 6 ($p = .000$). While all other LEVEL conditions did not differ significantly between each other in terms of length.

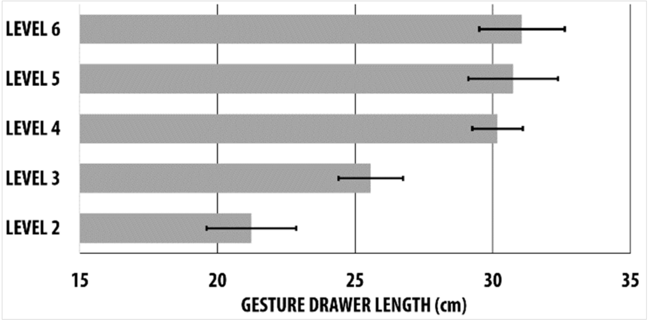

Figure 5: Mean user-defined *GestureDrawer* length for each LEVEL condition.

Figure 6 shows the distribution of selection points for each LEVEL and how the selection accuracy declines when the control size is getting decreased (higher LEVEL). We found a significant difference between the LEVEL conditions in terms of missed selections ($F_{4,68} = 32.781$, $p = .000$), see Figure 7. The post-hoc test revealed that the group of LEVEL 2, LEVEL 3 and LEVEL 4 and also the pair of LEVEL 5 and LEVEL 6 did not differ significantly between each other in terms of accuracy. When comparing the accuracy of different control positions (e.g. anchor position, end-point position, middle position) within each LEVEL no significant difference was found. We also investigated on which axis/axes combinations most missed selections occur ($x = 77\%$, $y = 9\%$, $z = 3\%$, $xy = 6\%$, $xz = 3\%$, $yz = 2\%$, $xyz = 1\%$).

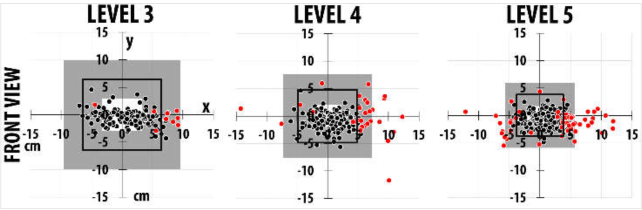

Figure 6: Interplay between accuracy and controls size (i.e. length). Correct selection points are encoded with black and incorrect with red dots. Black square indicates the average control size and gray filled square its standard deviation.

Measurement of selection time, from the point of the instruction till the point the participant made the selection, revealed a significant difference between the LEVEL conditions ($F_{4,68} = 7.889$, $p = .012$). The post-hoc test revealed all LEVEL conditions were significantly faster than the LEVEL 6 condition, while all other combinations were not significant. The average selection time for the group LEVEL 2 to LEVEL 5 was $\bar{x} = 1{,}993$ ms ($SD = 512$ ms) while LEVEL 6 had a selection time of $\bar{x} = 2{,}274$ ms ($SD = 502$ ms).

7.4 Discussion

As seen in Figure 4, *right*, participants mainly performed the *GestureDrawer* interaction technique a bit over the elbow height, slightly to the right from the shoulder position and in palm's reach in front of them. We would also like to point out the importance of depth (z-axis): we noticed that participants did not perform the opening action straight on the z-axis. Instead, their hand motions followed a curved path on which the controls were then positioned. In our study, the z-axis offset, between the anchor and end-point was on average -2.6 cm, which can be crucial once ignored (flat 2D GUIs for 3D gestural interaction), for example in LEVEL 5 the average control size was 7.6 cm and having a 2.6-centimeter precision was essential for participants' accuracy.

We found that the opening action length peaked at LEVEL 4 (average length of 30 cm), and stopped increasing after that. With this result, we have to reject *H1.1* since participants did not have a unique opening action length for each LEVEL. We argue that this peak is caused mainly by comfort. As the number of control items increased, participants tended to reach further away from the body up until a certain limit, where it became too uncomfortable for them to perform. Participants tended to interaction without moving their elbow or shoulder too much, similar reaction was also reported in [25].

Participants managed to place up to four (LEVEL 4) imaginary controls in 3D space using the *GestureDrawer* interaction technique and select them with an accuracy higher than 88%; despite having no visual feedback nor fixed landmark, see Figure 7. When the LEVEL was further increased, the selection accuracy started decreasing in a linear manner and was at 68% in our last condition (LEVEL 6). Therefore, we also can reject *H1.2* since the accuracy did not significantly decline before LEVEL 4 was reached. Figure 6 shows the interplay between accuracy and control size, where we can see that after the length stopped increasing after LEVEL 4, more controls were "squeezed" into the same *drawer* length; therefore, the control size decreased and the accuracy started declining (more red dots). Concluding, we found two accuracy ranges of the *GestureDrawer* interaction technique which are significantly different between each other, a high-accuracy range including LEVEL 2, LEVEL 3 and LEVEL 4 where participants' accuracy was on average 93% and a low-accuracy range with LEVEL 5 and LEVEL 6 with the average accuracy of 72%.

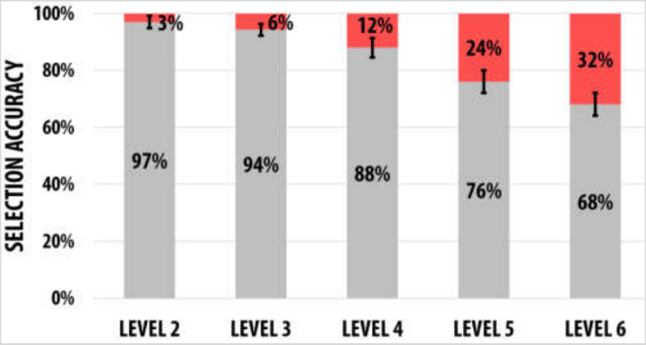

Figure 7: Selection accuracy of each LEVEL condition.

The selection time was the same for all LEVEL conditions except for LEVEL 6, for which we argue participants did not make their selection straight away as in all other LEVEL conditions. Instead they started to think about the control's position more intensively and this effect can be noted in the selection time.

The results of the control position comparison were also interesting. All position had the same accuracy. We expected that the anchor position and end-point positions would have a higher accuracy, as reported by related work with observable landmarks, where the accuracy is always higher the closer the participant's hand is to the landmark [12, 13, 24]. For our temporal reference frame created in the past by a clutch-based hand gesture, we could not report such findings.

8 STUDY 2: MEMORY RETENTION

In the second study, we investigated participants' abilities to maintain the visuospatial memory of the *GestureDrawer* while performing multiple control selection tasks, without reopening the *drawer* after each selection as in Study 1. Subsequently, we examined user performance under two conditions: visual feedback is given (via a GUI) or is absent.

8.1 Hypotheses

For this study, we had two hypotheses in mind:

- *H2.1*: A graphical user interface will provide more accuracy then an imaginary interface regardless of the number of controls in the *drawer*.
- *H2.2*: The control selection time will be shorter when visual feedback is absent.

8.2 Experiment Design

There were six conditions defined by the two-level independent variable SIGHTEDNESS and the three-level independent variable LEVEL. The LEVEL condition was identical to the first study, but we excluded LEVEL

2 and LEVEL 6 to keep the total study time shorter. SIGHTEDNESS consisted of a GRAPHICAL and IMAGINARY interface condition. In the GRAPHICAL condition, both the participants' hand (visualized as a 3D cursor) and the controls of the *drawer* were visualized in a 3D GUI on the apparatus display. In the IMAGINARY condition, participants did not receive any visual feedback (no GUI) and instead interacted fully by imagination.

During the study, each participant opened the *drawer* by using the *GestureDrawer* interaction technique for the adequate LEVEL. When the *drawer* was opened, the control-to-select was instructed. After the selection, the next control-to-select was instructed (in that time the participant was still holding the hand up, within the *GestureDrawer* interaction space). That way, participants sequentially selected each control three times without reopening the *drawer* and once completed, the next LEVEL was instructed. Once all LEVEL conditions in the first SIGHTEDNESS condition were completed, participants put their hands down and rested for two minutes. Afterwards the next SIGHTEDNESS condition was set. Selection correctness feedback was also provided by corresponding audio clips.

The study was a 2 SIGHTEDNESS (IMAGINARY, GRAPHICAL) × 3 LEVEL (LEVEL 3, LEVEL 4, LEVEL 5) within subject design. The presentation order of the SIGHTEDNESS and LEVEL conditions was counterbalanced across participants by using a balanced Latin square, and the control-to-select instruction was randomized. To understand the effect of memory retention, we also compared data from the first study, (where participants had to reopen the *drawer* after each selection and therefore always had a fresh memory "frame"), mentioned as IMAGINARY-REOPENED, against the SIGHTEDNESS conditions (where the *drawer* could not be reopened).

8.3 Results

Overall, we collected 1,296 target selections. All participants completed the study on average in 2 minutes 19 seconds ($SD = 49$ s).

We found that there is a significant difference between the two SIGHTEDNESS conditions ($F_{1,17} = 16.783$, $p = .001$) in terms of accuracy, see Figure 8, *left*. The post-hoc test revealed that the GRAPHICAL condition is significantly more accurate than the IMAGINARY condition ($p = .001$). Furthermore, it revealed a significant difference between the GRAPHICAL, IMAGINARY and IMAGINARY-REOPENED conditions ($F_{2,34} = 11.191$, $p = .000$). The post-hoc test revealed that the difference between IMAGINARY-REOPENED and GRAPHICAL was not significant in terms of accuracy ($p = .087$), while IMAGINARY was significantly less accurate compared to GRAPHICAL and IMAGINARY-REOPENED. Furthermore, we compared those three conditions separate within each LEVEL. We found that in LEVEL 3 there is no difference between the GRAPHICAL, IMAGINARY and IMAGINARY-REOPENED conditions ($F_{2,34} = .264$, $p = .769$), while in LEVEL 4 ($F_{2,34} = 12.428$, $p = .000$) and LEVEL 5 ($F_{2,34} = 9.803$, $p = .001$) there was a significant difference. The post-hoc test revealed that in LEVEL 4 the GRAPHICAL and IMAGINARY-REOPENED conditions were not significantly different ($p = .567$) while in LEVEL 5 there were significantly different ($p = .483$).

In LEVEL 3, where participants needed to sequentially perform 9 selections, the first error occurred on average after 8.2 selections, in LEVEL 4 with the total of 12 sequential selections after 6.3 selections and in LEVEL 5 with the total of 15 selections after 4.4 selections. From all missed selections in the IMAGINARY condition we also point out on which axis/axes combinations these errors occurred ($x = 48\%$, $y = 21\%$, $z = 4\%$, $xy = 16\%$, $xz = 6\%$, $yz = 1\%$, $xyz = 4\%$).

For the selection time, we found a significant difference between SIGHTEDNESS conditions ($F_{1,17} = 8.560$, $p = .009$) and LEVEL conditions ($F_{2,34} = 4.417$, $p = .020$) while the interaction between them was not significant ($F_{2,34} = 2.967$, $p = .065$), see Figure 8, *right*. The post-hoc test revealed that the IMAGINARY condition is significantly faster than the GRAPHICAL condition (p = .009) on average by 316 ms and that LEVEL 3 was significantly faster than LEVEL 4 ($p = .002$) and LEVEL 5 ($p = .000$) on average by 139 ms. Other combinations of both conditions were not significant.

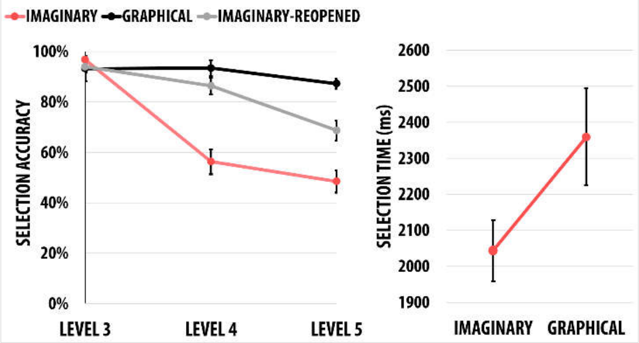

Figure 8: Sequential selection accuracy of SIGHTEDNESS conditions within each LEVEL (left) and control selection time of each SIGHTEDNESS condition (right).

8.4 Discussion

We found that the GRAPHICAL condition was in general significantly more accurate than the IMAGINARY, with the exception of LEVEL 3, where the accuracy did not differ. Thus, we can reject *H2.1* since the GRAPHICAL conditions was not more accurate in all LEVEL conditions. As reported previously [12, 30], we also found that a GRAPHICAL user interface resulted in visual search and had a negative effect on the selection time. Therefore, we can confirm *H2.2* since the IMAGINARY condition was indeed generally significantly faster than the GRAPHICAL.

We found that LEVEL 3 (and below), was the sweet spot for *GestureDrawer* sequential selection tasks, in which the IMAGINARY condition had the same accuracy and was simultaneously faster than the GRAPHICAL interface. In addition, LEVEL 3 was faster when compared to higher LEVEL conditions, which points out the high degree of confidence users had while performing the selections in it.

To investigate the effect memory degradation, we found that right after the *drawer* was opened in the IMAGINARY-REOPENED condition, the accuracy is indeed higher. Afterwards, in the IMAGINARY condition, the user's memory of the *drawer* starts fading, resulting an accuracy decline. We found that the accuracy of IMAGINARY-REOPENED in LEVEL 3, but more interestingly also in LEVEL 4, is not significantly different compared to the GRAPHICAL interface.

Therefore, we can conclude that (1) the *GestureDrawer* interaction technique used for IMAGINARY sequential selection tasks is generally faster compared to a gesture controlled GRAPHICAL user interface, (2) IMAGINARY is as accurate as GRAPHICAL when up to three controls are positioned in the *drawer*, and afterwards its accuracy declines below 60%, and (3) the accuracy may also be the same as in GRAPHICAL for LEVEL 4 if the *drawer* has been opened recently, and the user possesses a fresh imaginary "frame" of it (IMAGINARY-REOPENED).

9 STUDY 3: SPATIAL REVISITATION

In the third study, we investigated participants' revisitation ability and if the visuospatial memory acquired by *GestureDrawer* is robust enough to proprioception disturbance. In this study, we used a placement task (involving translation in the *x-y-z* directions) to lure each participant's hand away from the *drawer*, and then investigated if participants are able to revisit (i.e. navigate his hand back to) the initial control position once their hand has been multi-dimensionally shifted aside. The focus of this study was to investigate, if the revisitation ability is greater in lower dimensional placement than in higher ones.

9.1 Hypothesis

In this study, we explored the following hypothesis:

- *H3.1*: We expect that when the DOF of the placement task is increased, the harder it will be to revisit the initial control position, causing more reselection errors.

9.2 Experiment Design

There were three conditions defined by the independent variable DOF (1DOF, 2DOF, or 3DOF). Each of the three conditions represented an origin point of a one-, two- or three-dimensional Cartesian system, in which two objects needed to be aligned, see Figure 9. The 1DOF condition was used for one-dimensional placement tasks and described 6 rays *(+x, -x, +y, -y, +z, -z)*. The 2DOF condition was used for two-dimensional placement tasks and described 12 planes (*+x+y, +x-y, -x+y, -x-y, +x+z, +x-z, -x+z, -x-z, +y+z, +y-z, -y+z, -y-z*). The 3DOF condition was used for three-dimensional placement tasks and described octants of three-dimensional space (*+++, -++, --+, +-+, ++-, -+-, ---, +--*).

For this study, the *GestureDrawer* always contained three controls, each representing a DOF condition and its respective coordinate system. In a trial, the placement task was represented as a 3D GUI on the apparatus display, where the two 3D objects (cubes) of the same size (placer- and target-cube) needed to be aligned by translation, see Figure 9. The placer-cube, which the user needed to translate, was positioned on the origin point of the coordinate system of the instructed DOF (control position) and the target-cube was positioned with a certain off-set, on one or more axes (according to DOF). Before the study, we randomly generated an array of offset factors (between 0.3 and 0.7) for each ray, plane and octant, so that each participant had the same set of offset factors. In the study, we then multiplied this off-set factor with the user-defined *drawer* length and assigned it accordingly to the target-cube position based on the instructed DOF condition.

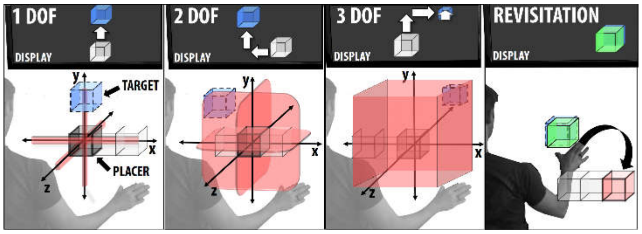

Figure 9: Examples of placement tasks for each DOF condition and the revisitation movement, for returning to the initial control position.

The participant firstly opened a *drawer* with three controls. A DOF was then instructed, and by a grasp gesture; the participant selected the control representing the instructed DOF and moved his hand while observing the placer-cube movement on the apparatus display, see Figure 9. The two cubes were considered aligned when their volumes overlapped each other by more than 70%. When this occurred, the color of the cubes would change to green and the participant could let go of the placer-cube. After that, a new pair of cubes would appear (placer-cube would be situated at the initial control position). Participants then needed to revisit (i.e. find the way back to) the initial control position once their hand was aside (due to the placement task) to select the new placer-cube.

Each ray, plane and octant position was instructed three times. This resulted in 78 trial (targets) per participant ((6 rays + 12 planes + 8 spaces) × 3 placement tasks). The presentation order of DOF was counterbalanced across all participants. For each participant, the DOF changed after all the targets were completed and the instruction order of targets was randomized. Participants rested for 2 minutes between changes of the DOF condition to prevent hand fatigue.

9.3 Results

In total, we collected 1,404 placement tasks. On average the study lasted 8 minutes 20 seconds (*SD* = 55 s).

The result analysis did not reveal a significant difference between the three DOF conditions in terms of the reselection accuracy of the instructed control ($F_{2,34} = .562$, $p = .569$), see Figure 10, *left*. We found that, independent of the DOF condition, in 70% of revisitation attempts participants correctly reselected the instructed control again. We also compared different axes, planes and octants in reference to the amount of missed selections they have caused and found no significant difference between the 6 axes in 1DOF ($F_{5,85} = .679$, $p = .580$), 12 planes in 2DOF ($F_{11,187} = 1.170$, $p = .328$) nor octants in 3DOF ($F_{7,119} = 2.108$, $p = .090$). Additionally, we found that once a participant missed the control selection after revisiting, it took them on average 1.75 (*SD* = 1.0) additional reselection attempts to find it and select it again.

For the placement time, we found a significant difference between the DOF conditions ($F_{2,34} = 33.209$, $p = .000$) and the post-hoc test revealed that all DOF conditions are significantly different between each other. Participants aligned the two cubes the fastest in 1DOF, then 2DOF and then in 3DOF as the slowest see Figure 10, *right*. Subsequently, we compared the time participants needed for the revisitation, and found that it is also significantly different between the DOF conditions ($F_{2,34} = 6.005$, $p = .011$). The post-hoc test revealed that only 1DOF was faster than 3DOF (p = .34), while 1DOF compared to 2DOF ($p = 1.000$) and 2DOF to 3DOF ($p = .064$) were not significantly different, see Figure 10, *right*.

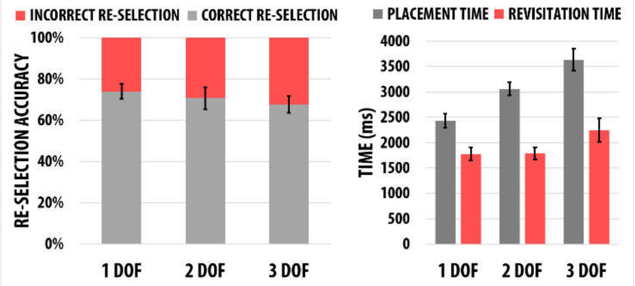

Figure 10: Selection accuracy of each DOF condition (left), placement and revisitation time of each DOF condition (right).

At the end of all three studies, we asked participants to rate how appropriate they found the grasp gesture as a clutching method (i.e. to step in contact with the system). They subjectively scored it with an average of 6.9 (*SD* = 2.3) on a scale from 0 to 10 (10 being very appropriate). Similarly, we asked about the opening action as a method for positioning and acquiring knowledge about imaginary controls self-positioned in empty space. Here the average score was 8.4 (*SD* = 1.5).

9.4 Discussion

We found no evidence that a higher dimensional placement task would negatively affect the revisitation accuracy more, than a lower dimensional placement task. The accuracy or our participants was in general around 70%, independent of the DOF condition and therefore we can reject *H3.1*. We argue, that participants did not limit their hand movement based on the axes available in each DOF, instead they considered all tasks as a 3DOF task, once they interacted entirely in 3D. Summarizing, we found no evidence suggesting that by limiting the number of dimensions for a particular control manipulation we would improve users' accuracy.

Subsequently we found that participants performed placement tasks with lower degree of freedom significantly faster and that the revisitation time did not follow the same trend, where 1DOF was significantly faster than 3DOF, but pairs 1DOF – 2DOF and 2DOF – 3DOF were not significantly different. We also found no evidence that movements into certain axes, planes or octants would negatively affect the participants' revisitation accuracy more than others.

We would also like to shortly discuss about the limitations we had in our implementation. We presented a 2.5D (pseudo-3D) placement task on a 2D display, where the depth was expressed by scaling, meaning that participants needed to do a minor cognitive remapping to associate their hand motion to the display content. Since we investigated participants' working memory and not cognitive workload, this did not influence our results. However, in future, other researchers may overcome this limitation by using mixed reality headsets instead.

10 IMPLICATIONS FOR DESIGN

Based on the three studies, we have gained first insights into the design space of one-handed gesture-controlled and user-defined 3D imaginary interfaces. We summarize design implications for such interfaces as follows:

1. Use the full 3D space for defining spatial user interfaces (including depth), for the benefit of user error prevention.
2. Avoid strong dependencies on physical devices, if you are creating simple interfaces for gestures and promote the design of spatially stable imaginary interfaces. This makes the user interaction cognitively easier, faster - reduces visual search, while keeping the same user accuracy as a graphical user interface.
3. Minimize interaction effort by designing gestures that simultaneously address different aspects of the interaction process, as the *GestureDrawer* opening action which embeds, self-definition addressing comfort, memory acquisition addressing learning, sophistication by gesture states addressing reliable detection.
4. Design short interaction cycles, since the user's memory starts degrading over time or make sure that is gets refreshed on time.
5. Use all directions in 3D space for manipulation of the user interface control. There is no notable difference that users would not be able to revisit the initial position of the control more accurately if we mathematically neglect the control's DOF, e.g. user rather one 2DOF panning tool, instead of two 1DOF sliders.
6. Consider the *GestureDrawer* interaction technique when you need a fast (<2000ms) and accurate (94%) access to a mid-air interface with shallow menu structures. For example, where users need a fast and accurate access to four controls without requiring their visual attention (e.g. in automotive, surgery, gaming).
7. Promote the underlying concept of imaginary interfaces when you use-case requires no device retrieval (while biking, driving or during meetings), one-handed interaction (while holding a phone, holding-on on a subway car), eyes-free interaction, when screens are not available or the graphical user interface real-estate is limited, when only fullscreen applications are preferred, for distant physical devices (e.g. data visualization walls, information displays), to lower the hardware cost needed for enabling user input (one gesture camera is enough) or for free-handed interaction (in sterile or anti-static rooms, car sharing).

11 APPLICATIONS

We implemented three demo applications that show the advantage of *GestureDrawer* in the automotive domain, see Figure 11.

Setting navigation pins on a map (3 controls): We open a *drawer* with three controls. Then we can grasp control one (representing the start pin) or control three (representing the destination pin), both of which are 2DOF controls. Once we grasp one of them, our hand movement is mapped to the selected pin and by moving our hand we can position the pin on the map (we release the grasp to confirm the position). When both pins are set, our route is determined. With control two (1DOF), we can now list between various alternative routes by moving our hand up or down.

Route time-travel (2 controls): While being driven to our destination, we open a *drawer* with two controls. We can look into the future of our trip, by grasping control two (1DOF) and by moving our hand forward or backward the navigation application (displaying a 360° street-view) starts scrolling the timeline of our route. Since only a part of the full 360° view can be displayed at once, we can we can grasp control one (2DOF), and by moving our hand we change the view accordingly. By that we can take a look at our points of interest and explore their surroundings.

Car's "home" drawer (4 controls): We can open a *drawer* with four controls to access the most used functions of our car. The first control represents the media player (2DOF), by which we can adjust the volume or changes the song, the second control represents air conditioning (2DOF) by which we can adjust the temperature (moving our hand up or down) or ventilation intensity (moving our hand closer or further away from our body), the third controls represents seat adjustment (2DOF), by which we can adjust our seat (up/down, forward/backward) and the forth controls turns the reading light on or off (0DOF - Switch). While we adjust any of these controls we receive output feedback by a screen-less output modality (i.e. volume/song - hearing, temperature/ventilation - thermoception, seat movement - vestibular sense, light - sight).

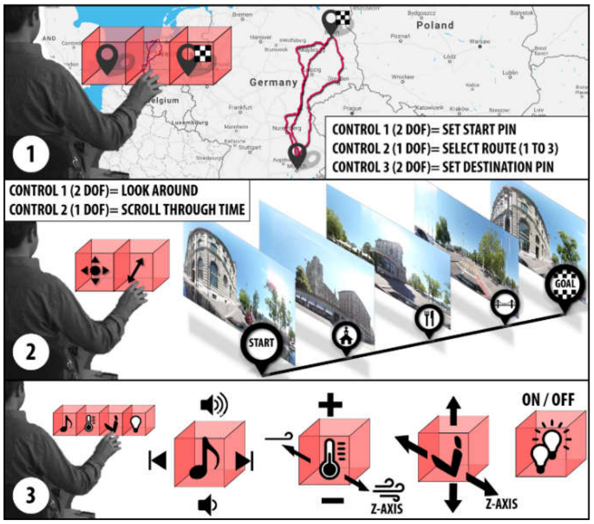

Figure 11: *GestureDrawer* applications: Navigation pins (1), Route time-travel (2) and Car's "home" drawer (3).

12 CONCLUSION

In this paper, we presented *GestureDrawer*, a novel one-handed interaction technique which allows users to place imaginary user interface controls in 3D empty space without receiving any kind of feedback (completely eyes-free) and without fixed landmarks (e.g. second hand). The validity of *GestureDrawer* was confirmed in three user studies, where it has been shown that by using it participants indeed instantly acquired enough knowledge to create an imaginary interface. This imaginary interface allowed them to locate its controls in empty-space and select or manipulate its functionalities by hand movements in all three dimensions. Furthermore, it was shown that with a limited amount of controls, participants interacted faster and as accurate using *GestureDrawer* than if they would use a gesture controlled GUI. Based on these findings, we developed design recommendations to leverage the use of a one-handed interaction technique for user-defined imaginary interfaces and demonstrated three applications addressing the automotive domain.

Through this work, we demonstrated the potential to improve gestural interaction by allowing users to have an influence on the definition of the user interface, avoiding long compulsory learning phases - to learn spatial positions of controls by muscle memory, reducing dependencies to physical devices and by promoting spatially stable interfaces which demand no visual attention. By bridging the gestural input directly to changes its actions cause in an "environment", without an intermediate GUI, we did a small step further from the visual paradigms (e.g. "*What you see is what you get.*") into the direction arguably more suited for post-WIMP interfaces, ideologically a more action-based paradigm [6], for example "*What you do is what you get.*" or "*What you do is what happens.*".

13 FUTURE WORK

As mentioned, *GestureDrawer* has two accuracy ranges; a high accuracy range (93%) allowing interaction for up to 4 controls and a low accuracy range (72%) for more than 4 controls, for this range we would consider applying reasonable feedback in the future to alleviate memory degradation in longer interaction cycles. Other future work possibilities include exploring combinations of imaginary and real-world interfaces, multiplexing controls, user-defined controls, usage of *drawer* length as a predictor for the number of controls and multi-user scenarios where people could share or exchange an imaginary interface (e.g. "*Pass me the remote for the ...*", "*The control for the … is in the first drawer.*").

REFERENCES

[1] Aigner, R., Wigdor, D., Benko, H., Haller, M., Lindbauer, D., Ion, A., Zhao, S., and Koh, J. Understanding mid-air hand gestures: A study of human preferences in usage of gesture types for HCI. Microsoft Research TechReport MSR-TR-2012-111 (2012).

[2] Baddeley, A.D., and Hitch, G. Working memory. The Psychology of Learning and Motivation: Advances in Research and Theory 8, (1974).

[3] Baudel, T., and Beaudouin-Lafon, M. Charade: remote control of objects using free-hand gestures. Commun. ACM 36, 7 (1993), 28-35.

[4] Bellotti, V., Back, M., Edwards, W.K., Grinter, R.E., Henderson, A., and Lopes, C. Making sense of sensing systems: five questions for designers and researchers. In Proc. CHI '02. ACM.

[5] BMW Gesture Control. 2017. https://arstechnica.com/cars/2016/01/playing-around-with-bmws-7-series-gesture-control-user-interface/ (2017).

[6] Bowman, D. A., Coquillart, A., Froehlich, B., Hirose, M., Kitamura, Y., Kiyokawa, K., and Stuerzlinger, W. 3D User Interfaces: New Directions and Perspectives. IEEE Comput. Graph. Appl. 28, 6 (2008), 20-36.

[7] Buxton, W. A three-state model of graphical input. In Proc. INTERACT '90. North-Holland Publishing Co.

[8] Cockburn, A., Quinn, P., Gutwin, C., Ramos, G., and Looser, J. Air pointing: Design and evaluation of spatial target acquisition with and without visual feedback. Int. J. Hum.-Comput. Stud. 69, 6 (2011), 401-414.

[9] Fuentes, C. T. and Bastian, A. J. (2010). Where is your arm? Variations in proprioception across space and tasks. Journal of Neurophysiology 103.

[10] Grosjean, J., Burkhardt, J.M., Coquillart, S., and Richard, P. Evaluation of the Command and Control Cube. In Proc. ICMI '02. IEEE.

[11] Gustafson, S. Imaginary interfaces: touchscreen-like interaction without the screen. In CHI EA '12. ACM.

[12] Gustafson, S., Bierwirth, D., and Baudisch, P. Imaginary interfaces: spatial interaction with empty hands and without visual feedback. In *Proc.* UIST '10. ACM.

[13] Gustafson, S., Holz, C., and Baudisch, P. Imaginary phone: learning imaginary interfaces by transferring spatial memory from a familiar device. In Proc. UIST '11. ACM.

[14] Gustafson, S., Rabe, B., and Baudisch, P. Understanding palm-based imaginary interfaces: the role of visual and tactile cues when browsing. In Proc. CHI '13. ACM.

[15] Hincapié-Ramos, J.D., Guo, X., Moghadasian, P., and Irani, P. Consumed endurance: a metric to quantify arm fatigue of mid-air interactions. In Proc. CHI '14. ACM.

[16] Hinckley, K., Pausch, R., and Proffitt, D. Attention and visual feedback: the bimanual frame of reference. In Proceedings of the 1997 symposium on Interactive 3D graphics (I3D '97). ACM.

[17] Hsieh, Y.T., Jylhä, A., Orso, V., Gamberini, L., and Jacucci, G. Designing a Willing-to-Use-in-Public Hand Gestural Interaction Technique for Smart Glasses. In Proc. CHI '16. ACM.

[18] Hürst, W. and Vriens, K. Multimodal feedback for finger-based interaction in mobile augmented reality. In Proc. ICMI 2016. ACM.

[19] Jude, A., Poor, M.P., and Guinness, D. Grasp, Grab or Pinch? Identifying User Preference for In-Air Gestural Manipulation. In Proc. SUI '16. ACM.

[20] Khambadkar, V. and Folmer, E. GIST: a gestural interface for remote nonvisual spatial perception. In Proc. UIST '13. ACM.

[21] Kulshreshth, A. and LaViola Jr, J.J. Exploring the usefulness of finger-based 3D gesture menu selection. In Proc. CHI '14. ACM.

[22] LaViola, J. Kruijff, E, Bowman, D., Pouyprev, I., McMahan, R. 3D User Interfaces: Theory and Practice (second edition), Addison-Wesley, 2017.

[23] Lin, S.Y., Shie, C.K., Chen, S.C., and Hung, Y.P. AirTouch panel: a re-anchorable virtual touch panel. In Proc. MM '13. ACM.

[24] Lin, S.Y., Su, C.H., Cheng, K.Y., Liang, R.H., Kuo, T.H., and Chen, B.Y. Pub - point upon body: exploring eyes-free interaction and methods on an arm. In Proc. UIST '11. ACM.

[25] Lubos, P., Bruder, G., Ariza, O., and Steinicke, F. Touching the Sphere: Leveraging Joint-Centered Kinespheres for Spatial User Interaction. In Proc. SUI '16. ACM.

[26] Nacenta, M.A., Kamber, Y., Qiang, Y., and Kristensson, P.O. Memorability of pre-designed and user-defined gesture sets. In Proc.CHI '13. ACM.

[27] Norman, D.A. Natural user interfaces are not natural. interactions 17, 2010.

[28] Norman, D.A., and Draper, S.W. User Centered System Design; New Perspectives on HCI. L. Erlbaum Assoc. Inc. (1986).

[29] Ruiz, J., and Vogel, D. Soft-Constraints to Reduce Legacy and Performance Bias to Elicit Whole-body Gestures with Low Arm Fatigue. In Proc. CHI '15.

[30] Scarr, J., Cockburn, A., and Gutwin, C. Supporting and Exploiting Spatial Memory in User Interfaces. Found. Trends Hum.-Comput. Interact 2013.

[31] Sherrington, C., 1907. On the proprio-ceptive system, especially in its reflex aspect. Brain 29 (4), 483–486.

[32] Song, J., Pece, F., Sörös, G., Koelle, M., and Hilliges, O. Joint Estimation of 3D Hand Position and Gestures from Monocular Video for Mobile Interaction. In Proc. CHI '15. ACM.

[33] Taylor, J., Bordeaux, L., Cashman, T., Corish, B., Keskin, C., Sharp, T., Soto, E., Sweeney, D., Valentin J., Luff, B., Topalian, A., Wood, E., Khamis, S. Kohli, P., Izadi, S., Banks, R., Fitzgibbon, A., and Shotton, J. Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences. ACM Trans. Graph. 35, 4, Article 143 (2016).

[34] Wang, C.Y., Hsiu, M.C., Chiu, P.T., Chang, C.H., Chan, L., Chen, B.Y., and Chen, M.Y. PalmGesture: Using Palms as Gesture Interfaces for Eyes-free Input. In Proc. MobileHCI '15. ACM.

[35] Wang, S., Song, J., Lien, J., Poupyrev, I., and Hilliges, O. Interacting with Soli: Exploring Fine-Grained Dynamic Gesture Recognition in the Radio-Frequency Spectrum. In Proc. UIST '16. ACM.

[36] Wigdor, D., and Wixon, D. Brave NUI World: Designing Natural User Interfaces for Touch and Gesture (1st ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[37] Yat Li, F.C., Dearman, D., and Truong, K.N. Virtual shelves: interactions with orientation aware devices. In Proc. UIST '09. ACM.