

# How To Make Large Touch Screens Usable While Driving

Sonja Rümelin<sup>1,2</sup>, Andreas Butz<sup>2</sup>

<sup>1</sup>BMW Group Research and Technology, Hanauerstr. 46 Munich, Germany, +49 89 38251985

<sup>2</sup>University of Munich (LMU), Amalienstr. 17, Munich, Germany  
{sonja.ruemelin, butz}@ifi.lmu.de

## ABSTRACT

Large touch screens are recently appearing in the automotive market, yet their usability while driving is still controversial. Flat screens do not provide haptic guidance and thus require visual attention to locate interactive elements that are displayed. Thus, we need to think about new concepts to minimize the visual attention needed for interaction, to keep the driver's focus on the road and ensure safety.

In this paper, we explore three different approaches. The first one is designed to make use of proprioception. The second approach incorporates physical handles to ease orientation on a large flat surface. In the third approach, directional touch gestures are applied. We describe the results of a comparative study that investigates the required visual attention as well as task performance and perceived usability, in comparison to a state-of-the-art multifunctional controller.

We found that direct touch buttons provide the best results regarding task completion time, but with a size of about 6x8 cm, they were not yet large enough for blind interaction. Physical elements in and around the screen space were regarded useful to ease orientation. With touch gestures, participants were able to reduce visual attention to a lower level than with the remote controller. Considering our findings, we argue that there are ways to make large screens more appropriate for in-car usage and thus harness the advantages they provide in other aspects.

## Categories and Subject Descriptors

H.5.2 [Information Interfaces and Presentation]: User Interfaces – *Graphical user interfaces (GUI), Haptic I/O, Input devices and strategies, interaction style.*

## General Terms

Design, Human Factors.

## Keywords

Direct touch, proprioception, haptics, physical objects, touch gestures, in-vehicle information systems, automotive user interfaces, visual distraction.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [Permissions@acm.org](mailto:Permissions@acm.org).

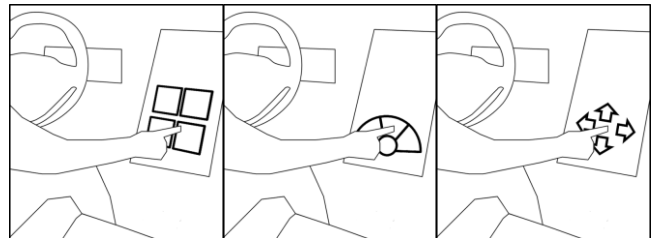
*AutomotiveUI '13*, October 28 - 30 2013, Eindhoven, Netherlands  
Copyright 2013 ACM 978-1-4503-2478-6/13/10...\$15.00.  
<http://dx.doi.org/10.1145/2516540.2516557>

## 1. INTRODUCTION

Recent trends in automotive displays, as exemplified by the latest Tesla Model S with its 17" touch screen<sup>1</sup>, show that in-car display spaces are becoming larger and incorporate an increasing amount of functionality. Touch screens have evolved beyond stand-alone navigation systems to sophisticated in-vehicle information systems, even integrating other functionality such as climate control.

Screen-based infotainment systems provide the advantage of direct control in contrast to the currently wide-spread remote-controlled UI concepts. This can result in shorter interaction times [10] [15]. Moreover, as the amount of functionality keeps increasing, screens can provide this functionality in a context-dependent way, for example by hiding climate control functions that are not required while a convertible is open.

However, touch screens lack the haptic nature of UI elements such as a radio button or a multifunctional knob, which let us find them by touch and can also indicate the result of an action blindly. This lack can have a negative effect on mental load and interruptibility [21].



**Figure 1. Touch interaction on large screens requires new interaction concepts to keep the visual attention on the road. Left: Enlarge interactive areas. Middle: Offer haptic guidance points. Right: Allow for position-independent touch gestures.**

In this paper, we present three approaches for improving interaction on large touch screens, using proprioception, haptic perception and position-independent touch gestures. To test their effect on task performance, visual distraction and subjective impression, we compared them in a driving simulator against a state-of-the-art remote controller interface. We found that the different approaches have specific effects on secondary task performance, while no negative effect on driving became apparent. In terms of task completion time, a direct touch interface is beneficial, while touch gestures have the potential to outperform a remote controller in terms of visual distraction.

<sup>1</sup> <http://www.teslamotors.com/models>

## 2. INSPIRATION AND RELATED WORK

Car manufacturers need to fulfill certain regulations regarding interaction and glance times [1]. Those apply to interaction with haptic controls as well as screen-based systems. In the following paragraphs, we present three concepts that inspired our design of new interface concepts to help conform to these regulations.

### 2.1 Direct Touch

Harvey et al. [15] compared indirect control with a rotary controller to the direct input on a touch screen, for 20 different secondary tasks, including music player interaction and climate control. They show that touch interaction consistently resulted in shorter interaction times and a higher usability rating, while driving performance remained stable. Usability issues they found were related to the high positioning of the 7" screen they used which led to fatigue. However, the UI design was optimized for touch control and thus did penalize the rotary controller.

There are approaches to add nonvisual feedback to touch screens, such as vibrations of the (whole) screen [13], pressure sensitive systems [24], remote tactile feedback [25] or electrovibrations [3]. These improve interaction in the lab, but it remains unclear how they will perform in the presence of driving vibrations in a car. Auditory feedback is useful for status confirmations, but does not help in targeting a certain interaction element on the screen.

When designing for large displays, their specific properties should be taken into account [4]. Proprioception, i.e., the sense of position and orientation of the body's parts with respect to each other, has been successfully used to identify different areas in the space around a user's body [19] and thus could also help to identify areas on a large screen. From Fitts' law [12] we know that pointing performance improves with increasing target size. Those results are not only true for pointer, but also for touch input [28].

### 2.2 Haptic Guidance

Ishii highlights the potential of Tangible User Interfaces to make use of the fact that "people have developed sophisticated skills for sensing and manipulating their physical environments" [16].

Pielot et al. [22] use the already available physical environment of a smart phone screen in a mobile application. They use the screen borders to ease orientation by guiding the interaction. Similarly, Rümelin et al. [26] compared dragging performance on a flat surface and along a horizontal bend, and found that the haptic support could ease interaction in terms of shorter task completion time and improved subjective impression. El-Glaly et al. [11] introduced physical elements on a touch screen to support spatial referencing for visually impaired people when using an e-reader, and found that it helps locating specific areas on the screen.

### 2.3 Touch Gestures

Bach et al. [2] compared three different input modalities in a music player task, and found that gesture interaction on a center console display leads to a reduced eye glance behavior, while direct touch enabled the fastest interaction. They compared gestures and direct touch with tactile button input, but did not take into account current remote-controlled interfaces.

Ecker et al. [9] present pieTouch, an approach to support blind navigation in hierarchical menus on a touch screen. They use a position-independent pie menu that is controlled by directional gestures. Blind interaction becomes possible when the positions of menu entries are known. They highlight drawbacks like

interruptability and the influence of car vibrations while performing the gestures.

Döring et al. [8] investigated the use of touch gestures on a steering wheel equipped with a multi touch display. In two tasks (map navigation and music player control), it caused less visual distraction than the physical controls of a standard radio and navigation system. Moreover, they found that gestures were easily learned due to experiences in other domains.

## 3. DESIGN PHASE

To investigate the effects of our three concepts, we integrated them into the design of interfaces for a 17" screen attached to the center stack, much like in the Tesla Model S.

### 3.1 Preliminary Considerations

Interactive areas of our prototype are located 20 to 35 degrees below the line of sight. Ergonomically, this is the range in which the driver can interact directly without removing the shoulder from the seat, which is a safety criterion. Moreover, we wanted test all interface variants on the same screen. One of them (*KnobTouch*) uses a haptic element in the middle of the screen. Therefore the other interfaces leave this small area blank.

### 3.2 Choice of Functionality

We refer to [2] and [8], who tested their concepts with a music player task. Listening to and controlling music is a common secondary task while driving. Adjustments are made on a regular basis. The subtasks we employ are shown in Table 1. We included *play*, *pause*, *skip song forward* and *skip song backward*. Due to the growing amount of playlist-based music players such as Spotify, or approaches to swipe covers as in iTunes, we think that navigation not only on a song-, but also on a playlist-level, will increase, so we added *skip playlist forward* and *skip playlist backward*. *Play* and *pause* are treated differently from the other subtasks. Because music can either be paused or play, standard music players such as Windows Media Player or iTunes alternate between those. We therefore needed to realize five actions.

**Table 1. Overview of subtasks when interacting with the music player.**

play	skip song forward	skip playlist forward
pause	skip song backward	skip playlist backward

### 3.3 Designing the Interfaces

Based on our analysis of related work, we wanted to investigate three different approaches to ease interaction on a large screen.

First, Fitts' law and the theory of proprioception suggest enlarging interactive areas to make them easy to hit, and identifiable only by the perception of muscle contraction and position of extremities. Therefore, our *SpaceTouch* interface uses touchable buttons with the maximum size (60 x 78 mm) that would fit our screen area.

Second, we attached a physical element, a turning knob, on the screen, to provide an anchor to grab for when interaction is initiated. In contrast to the screen's flat surface, it can be haptically identified. We then decided to align the touch buttons in equal distance around it to form a pie menu [6]. According to [9], we left out the area below the knob that is covered by the interacting hand, from 110° to 250° (clockwise, starting from the top). The button size in this second interface *KnobTouch* was also maximized to fit the screen (height 64 mm, max. width 70 mm).

Third, we designed a touch gesture interface, *SwipeTouch*. Gestures should be unambiguous to avoid errors in understanding, and should be easy to learn. The most commonly used touch gesture in consumer electronics is a swipe. A horizontal movement to the left or right is often used to switch between screens in mobile devices, so we decided to use that to switch songs and playlists. According to Pirhonen et al. [23], we chose a movement from left to right as the forward movement. To stay with a one-finger-gesture [18], we split the screen into two large areas: A horizontal touch gesture in the top area controls songs, while in the bottom area it controls playlists. A vertical swipe, performed anywhere on the screen, is used for the remaining functions. In preliminary tests we had found that moving the hand downward is a quickly performed gesture and thus well-suited to *pause* playing music. The contrary movement towards the top to *play*; however, was found to be a) uncomfortable to perform and b) irritating because *play* and *pause* are commonly understood as alternating functions (and implemented accordingly in other interfaces). Therefore, depending on the current music state, a downward touch gesture is triggering either *play* or *pause*.

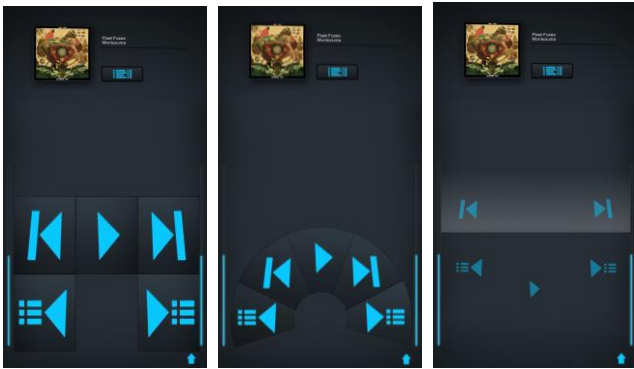


Figure 2. From top left to bottom right: *SpaceTouch*, *KnobTouch*, *SwipeTouch*.

Figure 2 shows the resulting interfaces. Two more interfaces were used for comparison. The first of them (Figure 3 left, *SmallTouch*) was meant to assess the effect of *SpaceTouch* compared to a standard touch interface. We used a button size of 30 x 30 mm, which was inspired by research of Colle and Hiszem [7] who recommend a touch button size of 20mm when using a kiosk standing in front of it. We further increased this size to compensate for car movements and vibrations.

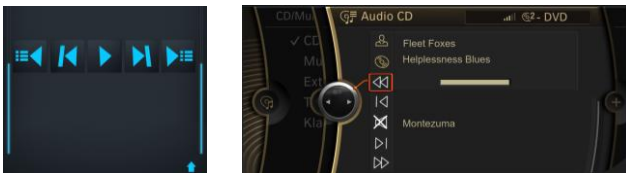


Figure 3. Left: *SmallTouch*, right: *RemoteControl*.

Moreover, we included a commercially available music player interface to compare the touch interfaces to. It was displayed in the top region of the center stack, and controlled with a remote multifunctional knob placed horizontally in the center console (Figure 3 right, *RemoteControl*). Functions are aligned vertically in a bar, with a pointer indicating the currently selected function. The knob can be turned left and right to navigate in the bar, and pressed to choose the current selection. If the knob is turned further when the last entry is reached, the pointer remains at the last position, and does not jump to the other end. The controller

can also be pushed towards left, right, top and bottom, but this is used for menu navigation, and not utilized for music player functionality.

## 4. SIMULATOR STUDY

We expected 1) that a spacious presentation eases interaction as no perfect direct targeting is required, 2) that a physical element can alternatively allow blind interaction on large touch screens by providing an orientation point, and 3) that it is possible to perform touch gestures without looking onto the screen. While our main focus was on visual distraction, we also investigated primary and secondary task performance, perceived workload, as well as overall usability and user preference.

### 4.1 Participants

40 participants (31 male, 9 female) with a mean age of 28 were recruited. Due to corporate confidentiality rules, all of them are working for the BMW Group, but were not involved in the current research. All of them have a driving license. 88% use touch screen displays of smartphones, tablets or ticket machines at least once a day. In contrast, only 15% use touch in cars daily (mostly integrated or attached navigation systems, but also their smartphones).

### 4.2 Apparatus

The experiment was conducted in the usability lab of BMW Research and Technology. The setup consisted of a steering wheel, instrument cluster, seat and pedals. The driving simulation was displayed on three 42'' screens in front of the participant. It consisted of a two-lane road where a front car was driving with a constant speed of 100km/h. Additionally, the setup was equipped with a multifunctional knob in the center console, and a 17'' touch screen in the center stack. On the screen, a (non-functional) knob of 42 mm diameter and 12 mm height was attached. For the interface using the remote multifunctional knob, only the upper part of the display was used, which corresponds to the display space available in current cars on the market. The application running the different interfaces as well as tracking the secondary task performance was developed with Adobe Flash CS5.



Figure 4. Experimental setup. A 17'' touch screen was attached to the center stack; additionally a multifunctional controller was integrated in the center console.

### 4.3 Experimental Design

As their main task, participants had to follow a car on a multilane road in a distance of 50m at a speed of about 100 km/h. For the secondary task, the music player control, a within-subjects design was used so all participants tested all five interfaces (*SpaceTouch*, *KnobTouch*, *SwipeTouch*, *RemoteControl*, *SmallTouch*). The order in which interfaces were tested was counterbalanced using a Latin square. For each interface, 18 tasks were performed. By task, we mean a single announced action in

the music player interface. The order was randomized, with the actions equally distributed. Two of them were requested three times in a row (e.g. “three songs forward”). As dependent variables, we measured visual distraction with a Dikablis eyetracking system, performance in the driving task as mean lateral and longitudinal deviation, performance in the music player task as task completion time and error rate, as well as subjective ratings for perceived usability (SUS), workload (NASA TLX) and experience (AttrakDiff).

#### 4.4 Study Procedure

The study started with the setup of the eyetracking system. Participants put on the eyetracking glasses and performed a calibration. Then the driving task was explained, followed by an accommodation phase in which they practiced following the front car. In the meantime, we verified that the eyetracking system was working properly. After that, participants were introduced to the prototype setup and the task of controlling the music player. They were presented with the first interface. An explanation of the functionality was given and they were asked to try out everything at least twice, until they felt familiar. After that, the driving simulation was started, and as soon as the car-following task was established, pre-recorded audio commands gave instructions to control the player. The researcher observed the task completion and took notes of unexpected occurrences. Afterwards, a questionnaire capturing subjective workload, perceived usability and user experience was completed. Then, the next interface was introduced. After all interfaces had been completed, a semi-structured interview was conducted to capture problems, preferences and general feedback. Overall, the study took about 75 minutes per participant. Everything was videotaped for later analysis.

#### 4.5 Results

Some datasets had to be excluded from further analysis because of problems with touch recognition we observed on the display we used. Results are based on the remaining, 3543 tasks performed. Though participants were asked not to include technical problems in their subjective ratings, this might have negatively influenced especially the touch-based interfaces. The results are reported at a significance level of .05.

##### 4.5.1 Task completion time

Task completion time was measured from the time the task was indicated, i.e. the voice command was given, to the successful selection. An ANOVA showed that task completion time of a single task is significantly influenced by the interface used ( $F(1,3135) = 5.8, r = .33$ ) (Figure 5).

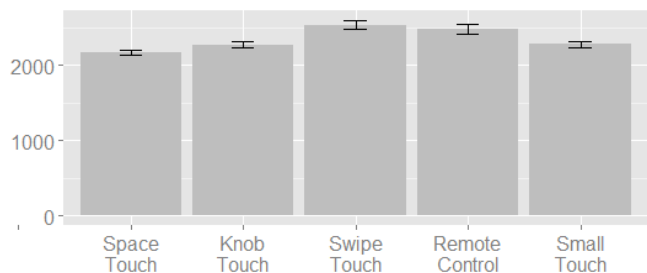


Figure 5. Mean task completion times for a single task (in ms). Error bars show 95% confidence intervals.

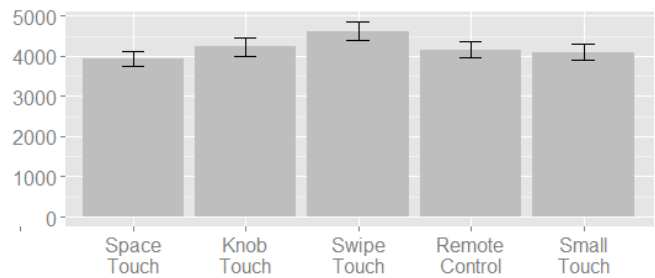


Figure 6. Mean task completion times for triple task selections (in ms). Error bars show 95% confidence intervals.

A post-hoc Wilcoxon test with Bonferroni correction reveals that using *SpaceTouch*, participants were significantly faster than with all others. In contrast, using *RemoteControl* or *SwipeTouch* extends the time needed significantly. There was a significant influence of the task that was performed for the remote controlled interface. *Play* always succeeded *pause*, and because those functions were alternating and at the same position in the function list (see Figure 3), the pointer was already in the correct position (*play*) after performing the *pause* task. As a result, *play* was performed significantly faster than *pause* for *RemoteControl* ( $F(1,78) = 27.5, r = .63$ ). When looking at the task completion time of tasks that included a triple selection of a function (e. g. “skip three songs”), a Friedman test still shows a significant influence of the interface used ( $F(1,350) = 8.9, r = .41$ ) (Figure 6). Pairwise comparisons only reveal an advantage of *SpaceTouch* over *SwipeTouch*.

##### 4.5.2 Error rate

Overall, the number of errors that occurred was low (Figure 7). *RemoteControl* only suffered from errors by choosing the wrong function, which might be due to an unclear, vertical alignment of functions, or an unsuccessful attempt to control it blindly. With *SwipeTouch*, comments make clear that errors are mainly caused by confusing the direction, because in our interface, one had to swipe to the left towards the icon “skip song”, whereas in cover flow visualizations, a swipe from the left brings in the next song/playlist. Looking at the error rate of the three different direct touch interfaces (*SpaceTouch*, *KnobTouch*, *SmallTouch*), the effect of the button size becomes apparent: the smaller the touch area, the more misplaced touches occur. Moreover, the size of the labels (i.e. the symbols of the player functions) depends on the size of the buttons and is therefore harder to read for smaller buttons. This might explain the increased number of wrongly chosen functions for smaller buttons.

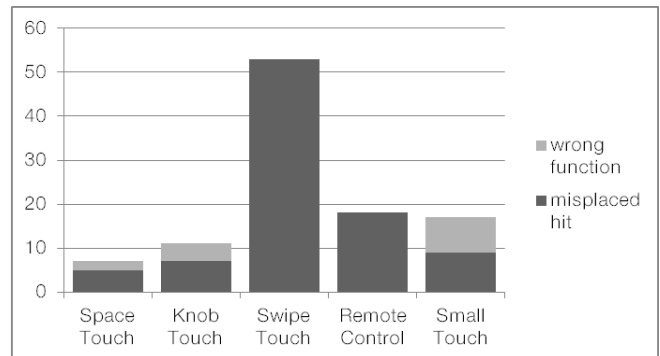
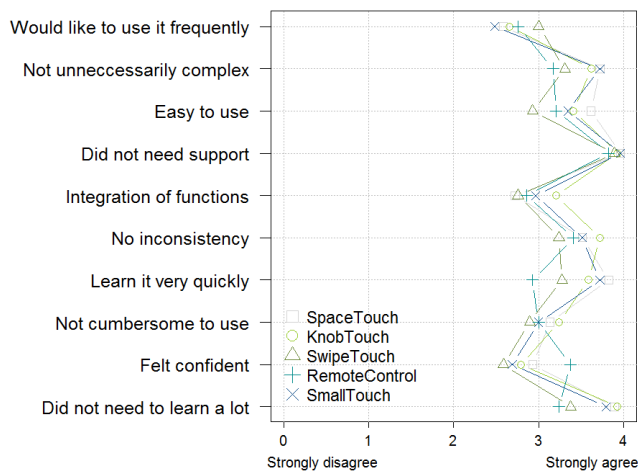


Figure 7. Overall number of errors for all participants, grouped into errors when a wrong function was chosen, and errors that resulted from misplaced hits in touch interfaces.



**Figure 8. Perceived usability ratings, assessed with the SUS questionnaire (Min. / low usability = 0, max. / high usability = 4)**

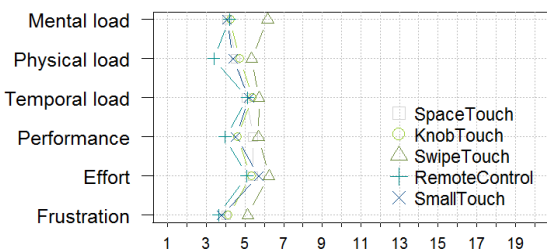
### 4.5.3 Usability

Perceived usability in ten different categories was assessed with the System Usability Scale (SUS) [5]. Figure 8 shows the results in each category. Overall, all systems were rated positively with mean ratings between 78 (*SwipeTouch*) and 86 (*SpaceTouch*, *KnobTouch*).

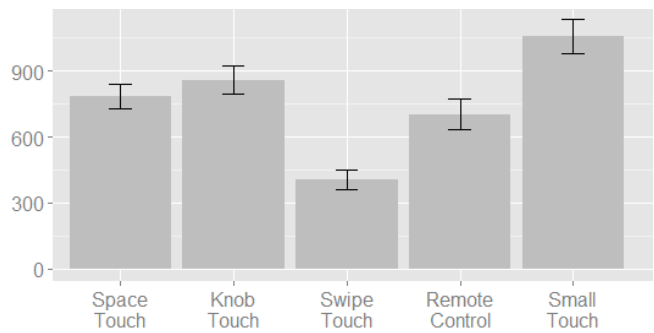
A Friedman test revealed a significant effect of the used interface on how quickly the system could be learned (7) ( $\chi^2(4) = 27.3$ ) and how much had to be learned (10) ( $\chi^2(4) = 31.2$ ). Post-hoc tests revealed that *RemoteControl* (7) (10) and *SwipeTouch* (7) were rated significantly worse than *SpaceTouch* and *KnobTouch*. Moreover, a higher complexity (2) is given for *RemoteControl*, compared to *SpaceTouch* and *SmallTouch* ( $\chi^2(4) = 14.7$ ). Regarding the perceived confidence, *RemoteControl* was rated significantly better than *SmallTouch* and *SwipeTouch* ( $\chi^2(4) = 12.9$ ).

### 4.5.4 Subjective Workload

Participants rated perceived workload with the NASA TLX questionnaire [14]. Figure 9 shows the results. Overall, all interface were rated to create a low to medium workload (25-34 of 120). The study setup may have influenced the impression of workload, as tasks followed each other in quick succession (10-15 sec). There was an effect of the used interface on physical demand ( $\chi^2(4) = 14.6$ ). *RemoteControl* was rated to be least exhaustive, but post-hoc tests show no significant results. Overall, *SwipeTouch* was assessed to be most demanding, no significant differences were found, too.



**Figure 9. Subjective workload ratings, assessed with NASA TLX questionnaire (Min. / low workload = 0, max. / high workload = 20)**



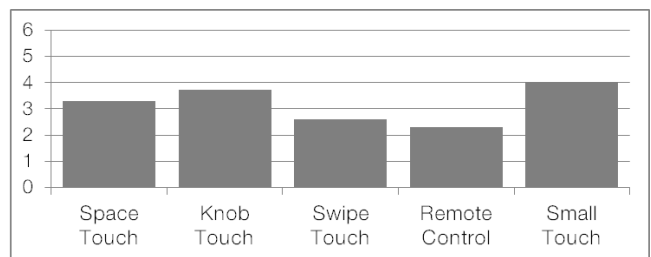
**Figure 10. Mean cumulated glance time per task (in ms) (n = 25). Error bars show 95% confidence intervals.**

### 4.5.5 Visual Distraction

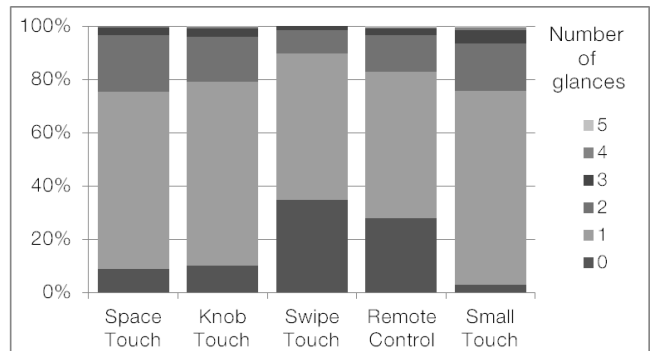
Visual distraction was measured with both questions on subjective impression and the collection of eye glance data with an eyetracking system for objective evaluation.

Due to tracking errors, some data sets had to be extracted for analysis; to keep a balanced experimental design, the data of 25 participants was used for analysis. The shortest mean glance duration of 403 ms was achieved using *SwipeTouch*, followed by 703 ms when using *RemoteControl* (Figure 10). An ANOVA ( $F_{4,1796} = 48.2$ ,  $r = .81$ ), and pairwise t-tests with Bonferroni correction revealed significant differences between all measurements except when comparing *SpaceTouch* with *RemoteControl* or *KnobTouch*.

Subjective ratings, presented in Figure 11, strongly support the results we achieved with the eyetracking data. Figure 12 depicts the distribution of the number of required glances. 34.7% of tasks with *SwipeTouch* were performed without a glance, in contrast to 28.0% with *RemoteControl*. *KnobTouch* has the highest number of tasks in which more than two glances were required, which corresponds with the longest mean glance time per task in Figure 10. However, it also showed the highest number of tasks performed without a glance of all direct touch interfaces.



**Figure 11. How often did you have to avert your eyes off the street? (0 = very rarely - 6 = very often)**



**Figure 12. Distribution of number of glances per task (n = 25).**



### 4.5.6 User Experience

The analysis of the results of the AttrakDiff questionnaire shows that the overall scores are located in the mid right area *task-oriented* (see Figure 13). Pragmatic quality (PQ), which explains whether or not the system fulfills functional goals by providing useful and usable means, is rated high. Hedonic quality (HQ), which describes whether it helps to fulfill individual needs, such as the desire to improve oneself or to communicate with others, was assessed to be between medium and high.

*SwipeTouch* shows a high HQ, presumably influenced by the new and most innovative modality, and because “*it is fun*”; however, it provoked many ideas for improvement and showed the worst technical performance. The good HQ of *KnobTouch* can possibly be explained by its visual design; several participants commented: “*it looks nice, aligned around the knob*”. *SpaceTouch* performed best in the PQ, possibly best explained by the large direct interaction. From a visual design side, it was the one to look less attractive, like “*...one of those mobile phones for elderly people*”.

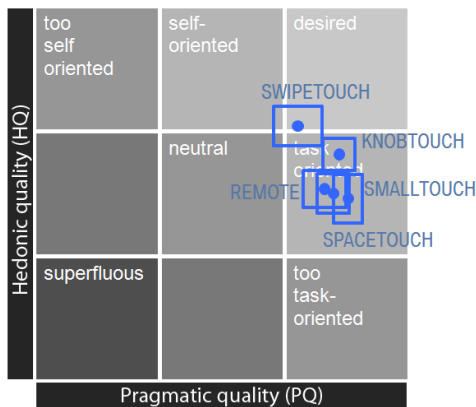


Figure 13. AttrakDiff. The participants rated the interfaces with contrary pairs of adjectives to evaluate the perceived hedonic and pragmatic quality.

### 4.5.7 Driving performance

Driving performance was measured with data taken from the driving simulation. Lane keeping was assessed with the mean lateral deviation from the road center [17]. No significant difference was found between the different interfaces ( $p > .05$ ,  $r = .10$ ). Moreover, the deviation from the optimum distance between simulator car and lead car was used to observe if drivers reduce their speed “in order to cope with the demand from the interaction with the secondary task” [15]. No significant effect could be found ( $p > .05$ ,  $r = .12$ ). Therefore, no negative influences of the secondary task on driving are apparent and we conclude that driving performance is not significantly reduced by any interface.

## 4.6 Discussion

### 4.6.1 Design of interfaces on a screen-based system

The presentation of information was not the same for directly and indirectly controlled interfaces. In the touch and gesture interfaces, the playlist cover was displayed in the top region, where the remote-controlled interface only presented artist, track and playlist name. Presenting information and functionality in a less cluttered way than in today’s remote-controlled systems is one potential of increased screen space. This had advantages for the tasks: For example, participants could see from the corner of the eye that an action had been successfully performed when the

cover changed. However, participants also commented that a colorful cover, or one they knew already, would sometimes attract their attention so it produced extra glances during the study, in addition to those needed for the task. Since we wanted to compare our touch interfaces to an existing system, we did not adapt the remote-controlled interface to our look and feel, which might have biased the results (in both directions).

What we found from comments and interviews is that the flexible design of screen-based systems is promising in a way that content and access can be tailored for different users, or for different scenarios, e.g., driving alone or with passengers. However, it has to be kept in mind that for certain functionality it makes sense to keep its position fixed, in order to harness spatial memory.

### 4.6.2 Direct touch is easier when large

The large touch buttons of *SpaceTouch* outperformed *SmallTouch* in terms of both task completion time and visual distraction (significantly shorter glance times). However, we made some remarkable observations. When we asked participants for their preference either while driving or when stationary, *SpaceTouch* was rated better in the dynamic condition ( $p < 0.01$ ), but not for the interaction in a standing car ( $p = 0.25$ ). This is supported by comments that highlight the usefulness of imprecise targeting while driving for *SpaceTouch*, but also that it is a waste of space when the driver can concentrate on the interaction. A solution to incorporate this feedback could be to adjust the button size to the driving situation or the current speed

Moreover, *SpaceTouch* was not favored from a design point of view, due to its bulky design. A solution could be to reduce the visible button size and show only the “iceberg” tip [27], or fade out borders to make them look valuable. Doing so, future work would have to investigate the effect of presenting smaller buttons with larger spacing on the impression of “*just having to tap in a certain area*”.

### 4.6.3 Direct touch for short interaction time

Looking at the task completion time, *SpaceTouch* outperforms *RemoteControl* significantly, with an average of 2168ms compared to 2478ms, a reduction of 12.5%. This coincides with the SUS results, where users rated the direct control of *SpaceTouch* to outperform *RemoteControl* in terms of perceived learnability and complexity. However, it required slightly more glance time, and there were only a small number of tasks performed blindly. This is supported by the significantly higher perceived confidence of the haptic feedback *RemoteControl* is providing. Overall, the size of the large touch buttons seemed to be not large enough, so users could not use their kinesthetic sense to “feel” where the buttons were located. Some participants, however, commented that using *SpaceTouch*, they could at least locate the region where the buttons are placed from the corner of their eye. After that, only a short glance was required to adjust their hand to the correct button. The general idea of orienting blindly in space was supported by the results of *SwipeTouch*, where the top and bottom region predominantly could be discriminated without looking.

The main problem for the touch interfaces is that the optimal areas to look at and to grasp for do not coincide. Therefore, there will always be a trade-off where to position the touchable elements. The few blind interactions mainly happened for the buttons in the bottom edges (22.2% of blind touches on previous / next song vs. 48.1% of blind touches on previous / next playlist), where users could, depending on their seating position, rest their arm on the

arm support in the center console and only move their arm from left to right in a very low position. For the buttons further up they had to orientate in “real 3D” space, which was also said to be potentially exhausting over a longer period of time.

#### 4.6.4 Does a physical anchor help orientation?

Compared to *SpaceTouch*, *KnobTouch* shows a slightly greater number of blind and one-glance interactions, despite the overall smaller size of buttons (28.5 cm<sup>2</sup> compared to 46.8 cm<sup>2</sup>). In return, this is accompanied with an increased task completion time, which might be caused by the time that is needed to locate the anchor. There are no significant effects, but it seems promising that single participants successfully tried to first approach the haptic element and then locate the respective touch button without looking at the screen. The current design of the physical element might partly account for that; with its height of 12 mm it seemed to be hard to find without touching the screen around it and thus most participants decided to make a control glance. In addition, when using *SmallTouch* or *SpaceTouch*, participants found their own physical orientation point. They grabbed the border of the screen with their hand while touching the outmost buttons with thumb or index finger. This indicates that for large touch screen designs, not only the touch area itself, but also the surrounding interior has to be taken into account.

#### 4.6.5 Touch gestures can be performed blindly

As found in previous research [2] [23], touch gestures like the ones used in *SwipeTouch* have the potential to be used without having to look at the screen. We confirm that they should be used for a limited function set to allow for simple, easy-to-perform gestures. Position-independent gestures like the up/down-swipe we used for *play/pause* have proven to prevent visual distraction, but we also showed that the function set can be extended by applying a simple gesture to different, sufficiently large areas. As discussed above, the top and bottom region we used were large enough to be identified blindly.

Compared to interacting with the multifunctional knob in the center console (*RemoteControl*), *SwipeTouch* showed significantly shorter glance times and a slightly higher amount of tasks that were controlled even without any glance. Participants commented that the matching between haptic feedback and cursor position required looking at the screen. To compensate, they developed different strategies to avoid visual distraction. For example, they remembered the last used function, or first moved the cursor to the very top and navigated blindly from there on, which was possible as the tasks were following in short distances briefly one after another. The significantly higher perceived confidence with haptic feedback of the knob could be due to the feedback we provided for the gestures. There only was a confirming signal when a gesture had been successfully completed. From the interviews we found that a constant feedback while performing the gesture could help in being more informed about the current status and thus increase confidence.

#### 4.6.6 Affordances of touch gestures

Despite the results of Pirhonen, we experienced several interaction errors when using *SwipeTouch*, which can be explained by a wrong understanding of the used direction. It seems that the mapping between directions and functions is strongly depending on former experiences. In our case, *skip forward* was related to a forward, left-to-right movement towards the respective icon, while *skip backward* was mapped to a backward, right-to-left movement. Participants mainly divided up

in two groups; those who have and those who have not had experience using Apple’s cover flow, in which a swipe in the opposite direction is required, as you “*fetch a cover from the right*” with a movement from to the left. This did not appear consistently, though; some participants mentioned that because of the different graphical representation, it was especially clear that the interaction was inverted. We conclude that with changing experiences of touch gestures in consumer electronics, interfaces have to be designed carefully and as robust as possible to the influence of similar use cases.

#### 4.6.7 Combining advantages of different modalities

As Bach et al. [2] and several participants suggest, direct and gesture-based touch interfaces can potentially be combined to provide a redundant access to functionality. Global touch gestures can be added as an overlay to touch interfaces, so the user can use the modality that fits best to the current situation. However, it would be required to indicate that touch gestures are possible unless they are used as expert functions that do not need to be apparent all the time. In that case they could be configured to control functions that are globally accessible. Apart from simple music player functions, that could provide access to certain functions such as switching between domains, for example a downward movement could open the player view, a movement to the right an overview of the traffic situation and so on. They would then serve as an entry point to different information screens in which further interaction is performed via direct touch.

## 5. CONCLUSION AND FUTURE WORK

In this paper, we have investigated different approaches to accessing functionality on a large touch screen and their effect on primary and secondary task performance.

We found that spacious direct touch interfaces have an advantage over remote-controlled interfaces in terms of task completion time and perceived learnability. Using a physical element, in our case a knob integrated into the screen with touch buttons aligned around it as a pie menu, did give a feeling of orientation in the large screen and allowed, in some cases, blind interaction. Incorporating physical objects in and around a large screen that can be identified by touch can help to maintain orientation in an otherwise flat interaction space. Touch gestures using directional movements outperformed controller interaction in terms of objective and subjective visual distraction while no difference in driving performance was found. The command set can be extended by applying the same gesture to different portions of a large screen. It is important that they are designed carefully to support the understanding of the linked functionality.

We could not find a significant effect of the interfaces on driving performance. The study was conducted in a driving simulator environment where users had to perform a car-following task. As a next step it will be important to conduct further experiments in a real driving setting to investigate further parameters that influence the suitability of the different interfaces for usage in the car.

Overall, our results show that different interaction concepts can keep up or even outperform the performance of a multifunctional controller, and aim to inspire further interface designers to make large touch screens usable - without taking visual attention away from the road.

## 6. ACKNOWLEDGEMENTS

We thank all participants for their time and their valuable feedback.

## 7. REFERENCES

- [1] Visual-Manual NHTSA Driver Distraction Guidelines. Technical report, Department of Transportation - National Highway Traffic Safety Administration, 2012.
- [2] Bach, K.M., Jæger, M.G., Skov, M.B., and Thomassen, N.G. You Can Touch, but You Can't Look: Interacting with In-Vehicle Systems. In *Proceedings of CHI '08*, ACM Press (2008), 1139–1148.
- [3] Bau, O., Poupayev, I., Israr, A., and Harrison, C. TeslaTouch: Electro vibration for Touch Surfaces. In *Proceedings of UIST '10*, ACM Press (2010), 283–292.
- [4] Baudisch, P. Interacting with Large Displays. *Computer* 39, 4 (2006), 96–97.
- [5] Brooke, J. SUS-A quick and dirty usability scale. In P. Jordan, B. Thomas, B. Weerd-meester and I. McClelland, eds., *Usability evaluation in industry*. Taylor & Francis, London, 1996, 189–194.
- [6] Callahan, J., Hopkins, D., Weiser, M., and Shneiderman, B. An empirical comparison of pie vs. linear menus. In *Proceedings of CHI '88*, (1988), 95–100.
- [7] Colle, H.A. and Hiszem, K.J. Standing at a kiosk: Effects of key size and spacing on touch screen numeric keypad performance and user preference. *Ergonomics* 47, 13 (2004), 1406–1423.
- [8] Döring, T., Kern, D., Marshall, P., et al. Gestural Interaction on the Steering Wheel – Reducing the Visual Demand. In *Proceedings of CHI '11*, ACM Press (2011), 3033–3042.
- [9] Ecker, R., Broy, V., Butz, A., and De Luca, A. pieTouch: a direct touch gesture interface for interacting with in-vehicle information systems. In *Proceedings of MobileHCI '09*, ACM Press (2009), 1-10.
- [10] Ecker, R., Broy, V., Hertzschuch, K., and Butz, A. Visual Cues supporting Direct Touch Gesture Interaction with In-Vehicle Information Systems. In *Proceedings of AutomotiveUI '10*, ACM Press (2010), 80–87.
- [11] El-Glaly, Y. and Quek, F. Touch-screens are not tangible: fusing tangible interaction with touch glass in readers for the blind. In *Proceedings of TEI '13*, ACM Press (2013), 245–253.
- [12] Fitts, P.M. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology* 47, 6 (1954), 381–391.
- [13] Fukumoto, M. and Sugimura, T. Active click: tactile feedback for touch panels. In *CHI '01 Extended Abstracts*, ACM Press (2001), 121–122.
- [14] Hart, S. and Staveland, L. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In P.A. Hancock and N. Meshkati, eds., *Human mental workload*. North Holland Press, Amsterdam, 1988, 139–183.
- [15] Harvey, C., Stanton, N. a, Pickering, C. a, McDonald, M., and Zheng, P. To twist or poke? A method for identifying usability issues with the rotary controller and touch screen for control of in-vehicle information systems. *Ergonomics* 54, 7 (2011), 609–25.
- [16] Ishii, H. Tangible bits: beyond pixels. In *Proceedings of TEI '08*, (2008), xv–xxv.
- [17] Knappe, G., Keinath, A., Bengler, K., Meinecke, C., and Alexander, F. Driving simulators as an evaluation tool - assessment of the influence of field of view and secondary tasks on lane keeping and steering performance. In *Proceedings of ESV '04*, (2004), 1–11.
- [18] Koskinen, H., Laarni, J., and Honkamaa, P. Hands-on the process control: users preferences and associations on hand movements. In *CHI '08 Extended Abstracts*, ACM Press (2008), 3063–3068.
- [19] Li, F.C.Y., Dearman, D., and Truong, K.N. Leveraging proprioception to make mobile phones more accessible to users with visual impairments. In *Proceedings of ASSETS '10*, ACM Press (2010), 187-194.
- [20] Moore, D.J., Want, R., Harrison, B.L., Gujar, A., and Fishkin, K. Implementing phicons: combining computer vision with infrared technology for interactive physical icons. In *Proceedings of UIST '99*, ACM Press (1999), 67–68.
- [21] Noy, Y.I., Lemoine, T.L., Klachan, C., and Burns, P.C. Task interruptability and duration as measures of visual distraction. *Applied Ergonomics* 35, 3 (2004), 207–13.
- [22] Pielot, M., Hesselmann, T., Heuten, W., Kazakova, A., and Boll, S. PocketMenu: Non-Visual Menus for Touch Screen Devices. In *Proceedings of MobileHCI '12*, ACM Press (2012), 327–330.
- [23] Pirhonen, A., Brewster, S., and Holguin, C. Gestural and Audio Metaphors as a Means of Control. In *Proceedings of CHI '02*, ACM Press (2002), 291–298.
- [24] Richter, H., Ecker, R., Deisler, C., and Butz, A. HapTouch and the 2+ 1 State Model: Potentials of Haptic Feedback on Touch Based In-Vehicle Information Systems. In *Proceedings of AutomotiveUI '10*, ACM Press (2010), 72–79.
- [25] Richter, H., Löhmann, S., Weinhart, F., and Butz, A. Comparing Direct and Remote Tactile Feedback on Interactive Surfaces. *Proceedings of Eurohaptics 2012*, Springer (2012), 301–313.
- [26] Rümelin, S., Brudy, F., and Butz, A. Up And Down And Along: How We Interact With Curvature. Presented at the workshop “Displays Take New Shape: An Agenda for Interactive Surfaces” in conjunction with CHI '13, 2013.
- [27] Saffer, D. *Designing Gestural Interfaces: Touchscreens and Interactive Devices*. O'Reilly Media, Sebastopol, Canada, 2008.
- [28] Sasangohar, F., MacKenzie, I.S., and Scott, S.D. Evaluation of Mouse and Touch Input for a Tabletop Display Using Fitts' Reciprocal Tapping Task. *Human Factors and Ergonomics Society Annual Meeting Proceedings* 53, 12 (2009), 839–843.



