A Study of Unidirectional Swipe Gestures on In-Vehicle Touch Screens

Gary Burnett, Elizabeth Crundall, David Large, Glyn Lawson Human Factors Research Group The University of Nottingham University Park, Nottingham +44 (0)115 9514040 {gary.burnett. elizabeth.crundall, dave.large, glyn.lawson}@nottingham.ac.uk Lee Skrypchuk Jaguar Land Rover Ltd Whitley Coventry +44 (0)2476 151497 Iskrypch@jaguarlandrover.com

ABSTRACT

Touch screens are increasingly used within modern vehicles, providing the potential for a range of gestures to facilitate interaction under divided attention conditions. This paper describes a study aiming to understand how drivers naturally make swipe gestures in a vehicle context when compared with a stationary setting. Twenty experienced drivers were requested to undertake a swipe gesture on a touch screen in a manner they felt was appropriate to execute a wide range of activate/deactivate, increase/decrease and next/previous tasks. All participants undertook the tasks when either driving within a right-hand drive, medium-fidelity simulator or whilst sitting stationary. Consensus emerged in the direction of swipes made for a relatively small number of increase/decrease and next/previous tasks, particularly related to playing music. The physical action of a swipe made in different directions was found to affect the length and speed of the gesture. Finally, swipes were typically made more slowly in the driving situation, reflecting the reduced resources available in this context and/or the handedness of the participants. Conclusions are drawn regarding the future design of swipe gestures for interacting with in-vehicle touch screens.

Categories and Subject Descriptors

H.5.2 User-Interfaces (Input devices and strategies, Interaction styles)

General Terms

Measurement, Performance, Design, Human Factors.

Keywords

Touch screens, Gestures, Driving, Distraction

AutomotiveUI 2013, October 27–30, 2013, Eindhoven, The Netherlands. Copyright 2013 ACM.

1. INTRODUCTION

Recently, there has been a move towards replacing traditional controls with touch screens in cars. The main functional benefit of using touch screens is that they provide the flexibility to display a large amount of useful information within a small space. However, the use of touch screens in cars does not come without disadvantages: Traditional controls such as knobs, switches and buttons are 'tactile' and can potentially be controlled without averting one's eyes from the road ahead, while touch screens have a uniform smooth surface and therefore typically require vision in order to operate them. While a driver is looking at a touch screen, they are not looking at the road or other driving-related stimuli, which is likely to have an adverse effect on steering [6], maintenance of lane position [5], and spotting hazards [1, 6] and lead to greater crash risk [8].

One way to avoid long off-road glances is to develop more eyesfree interfaces that can be operated without need for the visual modality. This has been achieved already on systems designed to be accessible by blind or partially-sighted people: Many touchscreen devices now ship with screen-reading software preinstalled. There is a diverse range of accessible touch screen interaction techniques, but most touch screens typically accept gestures as input and provide speech or audio as output. These gestures generally include swipes and taps to browse menus, predefined discrete gestures to perform actions (such as a swipe in a specific direction or drawing a shape), or gestures in specific regions of the screen (see [4] for a review). For example, Sanchez and Maureira [11] developed mBN, a mobile tool to assist blind and visually impaired people when navigating the subway. With this system, users interact with the hierarchical menus using a set of directional gestures between the corners of the screen (e.g. a downwards gesture on the right hand side for "next"; an upwards gesture on the left hand side for "quit").

Zhao et al. [17] developed an eyes-free menu technique using touch input and reactive auditory feedback. The 'EarPod' is used by making sliding gestures on a circular touchpad to provide access to hierarchical auditory menus. Users were as accurate with the EarPod as they were with a comparable visual technique and were faster with the former than the latter after 30 minutes of practice. Kane, Bigham and Wobbrock [3] developed 'slide rule', an eyes-free gesture technique with auditory feedback to improve touch screen accessibility for blind people. This is a multi-touch

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.

technique, which Kane et al. argue provides richer interactions, avoids the need to remember arbitrary gesture mappings, and provides access to more complex information. An important feature of this technique is that the user can drag their finger across the screen to scan items (with auditory feedback), then use a 2^{nd} finger tap to select their chosen item. Since no visual information is required on the screen (such as text labels), more menu items can be presented within a smaller amount of space.

The implementation of gestures within interfaces can reduce the need for visual feedback even in the case of sighted users. O'Neill et al. [9] developed a gesture input system combined with speech output and assessed whether the absence of a visual display impairs usability. The presence of a visual display had no benefit in terms of gesture accuracy. Furthermore, participants were actually slower with a visual display. Thus, the absence of a GUI display does not impair usability, at least for a system with speech output that provides a small number of semantically distinct services associated with memorable and distinct gestures.

When designing gestures for interfaces, several factors are going to influence ease of use. For example, how many different gestures should be employed? A system in which all commands are executed using gestures could be difficult to learn [16], and placing additional demands on a driver's memory could not only be frustrating but also have adverse effects on safety. Furthermore, what type of gestures should be employed? Gestures with certain characteristics (e.g. direction, length, speed) might be easier to carry out in certain contexts.

Gestures for use in touch screen interfaces are often defined by designers, with little or no input from end-users, so they are potentially biased by technological factors and not necessarily reflective of user behaviour. It has been shown that users often show preferences for gestures authored by large groups of potential end users to those created by a single designer [7]. Incorporating users in the design process is not a new idea and is evident in Participatory Design [12]. Wobbrock, Morris and Wilson [15] used this approach to investigate gestures used in surface computing. In their gesture-elicitation study, users were prompted with *referents* (or effects of an action), and were asked to perform signs (or causes of those actions). From this, Wobbrock et al. present a complete gesture set with agreement scores, and classify these gestures as a taxonomy. A similar approach has been used by Good et al. [2] to build a user-defined command-line email interface, by Wobbrock et al. [14] to develop EdgeWrite unistrokes, and by Ruiz et al. [10] to elicit end userdefined motion gestures with smartphones.

Using gestures for in-car touch-screen interfaces could reduce the frequency and duration of off-road glances, and have positive implications for road safety. Therefore, we used a similar gestureelicitation technique to Wobbrock et al. [15] to investigate the gestures instinctively used to carry out commands normally executed during driving, such as switching on the radio or decreasing the temperature. Gestures are likely to be influenced by the semantics of the command. For example, the directions of gestures prompted by commands to 'increase' temperature or volume might differ from directions of gestures which prompt the user to 'decrease' temperature or volume. Therefore, we were particularly interested in comparing different types of command. However, gesture preference might be also affected by aspects of driving and the car environment. For example, a touch screen mounted on the central console might have to be operated using the driver's non-preferred hand, which might bias drivers towards using particular gestures over others. Therefore, we compared gestures used in a static situation (i.e. where the touch screen was held in a preferred position whilst seated and operated with either hand) with gestures used while driving in a car simulator in order to find out whether aspects of driving influence gesture preference.

In this experiment, we specifically looked at unidirectional swipe gestures from a fixed point to make it simpler to draw comparisons between different conditions. Although investigation of multidirectional gestures from flexible, user-defined starting points no doubt warrants investigation, this is beyond the remit of this particular experiment. Gesture recognition algorithms use a set of features to classify user input (Tu, Ren and Zhai, 2012), of which gesture length, direction and speed are relevant to unidirectional swipe gestures made from a fixed point. Therefore, these three measures were used to compare gestures in different conditions.

2. METHOD

2.1 Participants

Participants were recruited from the University of Nottingham (staff and students), and 20 people (9 males, 11 females) took part in the study. Participants' ages ranged from 20 to 55 years old (mean age = 32 years; s.d. 10 years). Six of the participants were left-handed and 14 were right-handed. All participants had been driving for at least two years. Six participants were iPhone users, ten were Android users, and four used other types of phone. Seven used iPad tablets, two used android tablets and one used a BB Playbook.

2.2 Design

There were two repeated measures variables. The first was the type of command, of which there were six categories: 'Activate', 'Deactivate', 'Increase', Decrease', 'Next' or 'Previous'. The order in which commands were delivered was completely randomised and not blocked by type. The second independent variable was the 'context' in which the participant made their gesture. This had two levels: 'Static' and 'Driving'. Presentation of Static and Driving conditions was counterbalanced. Dependent measures were direction of gesture, gesture length and gesture speed.

2.3 Apparatus and Stimuli

2.3.1 Verbal Commands

A list of 32 different commands was prepared, to be presented verbally by the experimenter. Each command asked the participant to perform a typical in-car or driving-related activity such as changing a setting or moving through a list. Commands were categorised a priori as one of six different types. The six types differed in terms of the type of change or movement implied by the command and the direction of this change or movement. Increase/Decrease commands all implied that an alteration should be made to the intensity of a setting. Next/Previous commands all implied moving forwards or backwards. Activate/Deactivate commands all implied starting or stopping. The 32 commands are arranged by category in Table 1.

Activate	Increase	Next	
Play music	Make music louder	Play next music track	
Switch on air	Make fan speed	Move from A to Z	
conditioning	quicker	in music list	
Answer telephone	Speed up fan	Go to next navigation instruction	
Switch on	Make temperature	Move from A to Z	
navigation	warmer	in phone contacts	
	Make navigation messages louder	Fade music to front of car	
		Scan to next radio station	
		Fast forward through music track	
Deactivate	Decrease	Previous	
Pause music	Make music quieter	Play previous music track	
Stop music	Make fan speed slower	Move from Z to A in music list	
Switch off air conditioning	Slow down fan	Replay navigation instruction	
End telephone	Make temperature	Move from Z to A	
conversation	cooler	in phone contacts	
Switch off	Make navigation	Fade music to back	
navigation	messages quieter	of car	
		Scan to previous	
		radio station	

Table 1. Categorisation of commands

2.3.2 Touch Screen

Participants made their swipe gestures on an iPad 2, which was always orientated with the home button to the right. The iPad was either held by the participant at a comfortable distance (Static condition), or positioned in the centre console of the car to the left of the driver (Driving condition). The screen was completely black apart from a green hotspot in the centre. The purpose of the hotspot was to control the starting point of gesture.

The interface was written in html/php and delivered via a wireless internet connection. The gesture start and end locations, and gesture start and end times were captured using html/php (timings were captured on client side and passed to server side php script where they were saved to a text file to ensure that delays in network transfer did not affect the results).

2.3.3 Simulator

The driving condition took place in a fixed-based, mediumfidelity driving simulator. The simulator comprised the front half of a 2001 right-hand drive Honda Civic SE car positioned within a curved screen providing approximately 270° viewing angle. The driving scenario was projected onto the screen using three overhead projectors, with rear views relayed to the side mirrors using video cameras and LCD displays. A fourth overhead projector was used to project the rear view to a screen situated behind the car, which could be seen by the driver using the existing rear-view mirror.

Drivers were able to interact with the car and driving scenario using an authentic steering wheel which provided force feedback, accelerator, brake and clutch pedals and steering column controls, such as indicators, situated within the car. The simulated driving scenario and driving experience were created using STISIM (version 2) software. A bespoke Java application was integrated with the STISIM software to calculate road speed; this was presented on an 8 inch LCD display fitted into the instrument panel to mimic the car dashboard.

2.3.4 Driving Scenario

The driving scenario comprised a single carriageway with fields and trees to both sides to indicate a rural setting. The road itself was winding, and included several sharp bends. There was also substantial traffic behind the participant's vehicle and approaching in the opposite lane. A challenging scenario was used to ensure that participants had to control the vehicle at all times and gave priority to the primary task of driving. This also ensured that participants responded instinctively to the commands and did not have much time to deliberate their responses.

2.4 Procedure

Ten of the twenty participants took part in the Static condition first and the other participants took part in the Driving condition first. In the Static condition, participants were seated away from the driving simulator. In the Driving condition, the participant was seated in the driver's seat of the simulator in their normal driving position. Participants drove in the simulator for approximately ten minutes before they were required to interact with the touch screen, so that they could get used to the controls and the driving scenario.

In each condition, participants were given 32 verbal commands delivered consecutively in random order by the experimenter. Participants were asked to respond to each command by making a single swipe gesture on the touch screen, which they instinctively felt was appropriate for executing that particular command. In the Static condition, participants were free to make their gestures using their preferred hand. In the Driving condition, due the position of the touch screen relative to the steering wheel, participants were required to make their gestures using their left hand.

Participants were informed that each gesture had to start in the green hotspot in the centre of the screen and should be a continuous movement using one finger, with the finger remaining in contact with the screen at all times during their chosen gesture. Since there was no visual feedback, participants were advised that when carrying out their gesture, they should assume to be already in the correct menu, or environment, to carry out that command. It was made clear that they did not need to make each gesture different and that they should therefore try to avoid remembering previous responses. After completing a gesture, the central green spot disappeared and the screen went completely black. After 5-10 seconds, the central green spot reappeared and the next command was given by the experimenter until all 32 gestures had been completed.

3. RESULTS

Participants' gestures were analysed in terms of direction, length and speed. In order to analyse gesture direction, the screen was divided into 8 sections as shown in figure 1, and each gesture was labelled as being 'left', 'right', 'up', 'down', 'up-left', 'up-right', 'down-left' or 'down-right' on the basis of the angle of the gesture. Gesture length was calculated as the linear distance from the gesture start point to the gesture end point in pixels. Gesture speed was calculated as the length of the gesture divided by its duration, and is therefore given in pixels per second.



Figure 1. Directions used in coding of gestures

3.1 Gesture Direction

3.1.1 Consensus

In analysis, an initial consideration was whether consensus occurred for any tasks related to the direction of swipe made. In this respect, Table 2 highlights the tasks (and associated gesture directions in brackets) in which either 80% (16 of 20) or 60% (12 of 20) of participants made a swipe gesture of the same direction for the Driving condition.

Table 2	2.	Consensus	in	swine	direction
1 4010 2	•	Comsensus		Supe	anection

	Tasks that meet the threshold (with associated direction in brackets)		
80% threshold	Make music louder/quieter (up/down)		
(16 out of 20 participants)	Play next/previous music track (right/left)		
	Fast forward/backwards through music track (right/left)		
	Scan to next/previous radio station (right/left)		
60% threshold	Make music louder/quieter (up/down)		
(12 out of 20 participants)	Make temperature warmer/cooler (up/down)		
	Make navigation messages louder/quieter (up/down)		
	Play next/previous music track (right/left)		
	Play next/previous navigation message (right/left)		
	Fast forward/backwards through music track (right/left)		
	Scan to next/previous radio station (right/left)		
	Fade music to front/back of car (up/down)		

3.1.2 Direction preferences

Direction of gestures made in response to 'Activate/Deactivate', 'Increase/Decrease' and 'Next/Previous' commands were analysed in three separate sets of tests. In each set, two Chi Squared tests were performed, one test using data from the Static condition and one test using data from the Driving condition. Each test compared the frequencies of gestures made in the eight possible directions for the two different types of command (activate vs. deactivate; increase vs. decrease; or next vs. previous).

Analysis of activate/deactivate gestures revealed that gesture direction was dependent on command type in both Static and Driving conditions (Static: $\chi^2_{(7)}=96.6$; Driving: $\chi^2_{(7)}=64.6$; p<.001 in both cases). Standardised residuals revealed that participants made more rightward and upward gestures, and more diagonal gestures towards the top-right, in response to 'activate' commands. In contrast, participants made more leftward and downward gestures in response to 'deactivate' commands (all standardised residuals > 2). This was the case for both Static and Driving conditions.

Similar results were found in the analysis of increase/decrease gestures. In both Static and Driving conditions, the Chi Squared results were statistically significant (Static: $\chi_{(7)}^2$ =178.3; Driving: $\chi_{(7)}^2$ =191.1; p<.001 in both cases). In both Static and Driving conditions, standardised residuals revealed that participants made more rightward and upward gestures, and more diagonal gestures towards the top-right, in response to 'increase' commands. In contrast, participants made more leftward and downward gestures, and diagonal gestures towards the bottom-left, in response to 'decrease' commands (all standardised residuals > 2).

The analysis of next/previous gestures also revealed that gesture direction was dependent on command type in both Static and Driving conditions (Static: $\chi^2_{(7)}=121.7$; Driving: $\chi^2_{(7)}=146.4$; p<.001 in both cases). Standardised residuals revealed that participants made more rightward gestures in response to 'next' commands, and made more leftward gestures in response to 'previous' commands (all standardised residuals > 2). This was the case for both Static and Driving conditions.

A further six separate Chi squared tests were performed on gestures made in response to 'activate', 'deactivate', 'increase', 'decrease', 'next' and 'previous' commands, each test comparing the frequencies of gestures made in the eight possible directions for the two different contexts (Static vs. Driving). However, there was no evidence that the direction of gestures for any of the different types of command were dependent on context.

Figures 2-4 show the percentage of gestures that were made by participants across both Driving and Static conditions, in each of the eight directions, for activate/deactivate, increase/decrease and next/previous tasks.



Figure 2. Directions of gestures made for activate/deactivate tasks



Figure 3. Directions of gestures made for increase/decrease tasks



Figure 4. Directions of gestures made for next/previous task

3.2 Gesture Length

A repeated measures ANOVA was performed on gesture lengths, comparing the six levels of Command Type (Activate; Deactivate; Increase; Decrease; Next; Previous) and the 2 Context conditions (Static; Driving). There was a significant effect of command type, F(5,95) = 24.56; MSE = 3746.78; p<.001. Pairwise comparisons revealed that gestures for 'deactivate' commands were significantly shorter than gestures for 'activate' commands (p<.01), and in both cases, these gestures were significantly shorter than gestures for 'decrease' command types (max. p<.01). Gestures for 'decrease' commands were significantly shorter than gestures for 'increase', 'next' and 'previous' commands (all p<.05) – see Figure 5. No other effects reached statistical significance.



Figure 5. Mean gesture length (in pixels) for each command type in Static and Driving conditions.

3.3 Gesture Speed

A repeated measures ANOVA was performed on the speeds of gestures, comparing the six levels of Command Type (Activate; Deactivate; Increase; Decrease; Next; Previous) and the 2 Context conditions (Static; Driving). There was a significant effect of command type, F(5,95) = 4.52; MSE = 38435.07; p<.01. As can be seen in Figure 6, the slowest gestures were in response to 'decrease' commands, and pairwise comparisons revealed that these gestures were significantly slower than all other gestures (max. p<.05) apart from 'deactivate' gestures. 'Previous' gestures were the fastest, and were shown to be significantly faster than 'deactivate', 'increase' and 'decrease' gestures (max. p<.05).

There was also a significant effect of context, F(1,19) = 6.68; MSE = 115589.91; p<.05, which showed that gestures were faster in the Static condition (mean = 681.7 pixels/s) than in the Driving condition (mean = 568.3 pixels/s). However, there was no significant interaction between context and command type (p=.676).



Figure 6. Mean gesture speed for each command type in Static and Driving conditions.

3.4 Effects of direction on length and speed

It is possible that some of the effects of command type on gesture length and speed are confounded by gesture direction. For example, 'increase' gestures were faster than 'decrease' gestures, but the former tended to be rightward movements and the latter tended to be leftward movements, so it is possible that rightward movements are simply faster. It is also possible that any effects of direction on length and speed are dependent on context. Therefore, 2 separate 2-way repeated measures ANOVAs were performed, one on gesture length and the other on gesture speed, with gesture direction (right; left; up; down) and context (Static vs. Driving) as independent variables. The 4 other gesture directions (up-right; up-left; down-right; and down-left) were excluded as there were so few of them, and many participants did not make any gestures in these directions. Despite this, two participants were excluded from the analyses because they did not make gestures in all four directions in both contexts.

Table 3 shows summary statistics for different directional gestures in Static and Driving conditions. There was a significant effect of direction on length of gestures, F(3,51) = 9.78; MSE = 3835.51; p<.001. Both upwards gestures and downwards gestures were significantly shorter than leftwards and rightwards gestures (max. p < .05), and downwards gestures were also significantly shorter than upwards gestures (p < .01). There was also a significant effect of direction on speed of gestures, F(3,51) = 9.23; MSE = 35835.58; p<.001). Downwards gestures were significantly slower than all other gestures (max. p<.05), and upwards gestures were also slower than leftwards gestures (p<.01). There was also an effect of context on gesture speed, which merely reflected the same effect found in the previous analyses and showed that gestures were faster in the Static condition than in the Driving condition. There were no interactions between context and direction, so there is no evidence to suggest that the effects of direction on gesture length and speed are altered by context.

Table 3. Mean length and speed of rightward, leftward, upward and downward gestures in Static and Driving conditions, with standard errors in parentheses

		RIGHT	LEFT	UP	DOWN
Static	Length (pix)	298.3	314.0	287.3	245.6
		(19.0)	(23.2)	(16.1)	(14.5)
	Speed (pix/s)	695.6	709.7	697.3	542.4
		(98.3)	(91.7)	(99.8)	(82.5)
Driving	Length (pix)	312.6	321.1	258.7	246.8
		(16.3)	(20.4)	(18.7)	(13.8)
	Speed (pix/s)	654.3	702.7	512.0	437.1
		(82.3)	(96.5)	(58.1)	(54.7)

4. DISCUSSION AND CONCLUSIONS

This study considered how the direction, length and speed of swipe gestures for a touch screen varied for different in-vehicle tasks and between the driving and static situations. Swipe gestures are of particular interest in the driving context because of their potential to significantly reduce visual demand when compared with traditional on-screen buttons.

Table 2 concerns the level of agreement offered by participants in the direction of swipe made. The table highlights the potential coding that designers could use for a swipe direction for common in-vehicle tasks, according to the level of confidence required. A lower consensus level would be associated with a greater likelihood of error in the initial direction of swipe made by a user population. Nevertheless, a 60% threshold clearly provides more opportunities for designers to code in-vehicle tasks with swipe directions. In addition, on-screen cues or training procedures could be adopted to assist drivers in learning the correct gesture for a given task.

In observing Table 2, it is clear that the highest agreement in the swipes undertaken occurred for the tasks which are likely to be most familiar to users of touch screen smartphones, that is, those associated with the playing of music (increasing/decreasing volume, next/previous track). In these cases, it is highly likely that stereotypes have been formed which can be transferred over to the driving environment. Moreover, the probability is that a wider range of stereotypes will develop as touch screens are adopted within alternative contexts.

To highlight the consensus found for music tasks, Figures 7 and 8 show the screen recordings of swipes made for all 20 participants for two specific tasks, "Making the music louder" and "Play previous music track".



Figure 7. Swipes made for "Making the music louder" task



Figure 8. Swipes made for "Play previous music track" task

A further observation from Table 2 is that consensus is apparent for certain increase/decrease and next/previous tasks, but not for activate/deactivate tasks. Two factors are of relevance here. Firstly, increase/decrease and next/previous tasks have clear spatial/location-oriented content which can be mapped on to a swipe direction. Secondly, activate/deactivate tasks are less likely in current touch screen interfaces to be associated with a swipe gesture. Indeed, several participants commented that they were used to conducting these tasks by tapping on-screen buttons. Whilst absolute consensus did not exist for activate/deactivate tasks using an 8-way split in direction, it is clear from the statistical analysis in section 3.1.2 and Figure 2 that consistent differences exist. In particular, participants tend to associate an activate task with a right/upwards swipe and a deactivate task with a left/downwards swipe. What this result highlights is the need for different splits in the coding schemes for gesture direction, based on the precision with which a user population may distinguish between gesture directions. For increase/decrease and next/previous tasks, it is possible that 8-way splits in direction (or possibly 4-way) could be utilised. This would enable a designer to code a swipe such that several commands could potentially be executable from a given starting point (e.g. a whole screen or part of a screen). For instance, in a music play mode, a swipe to the right/left could move to the next/previous track, whereas a swipe up/down could increase/decrease music volume.

In contrast, for activate/ deactivate tasks, a simpler two-way differentiation may be desirable, where a right/up swipe turns a function on (e.g. AC), and a left/down swipe turns that same function off. Indeed, if one considers Figure 2, it is apparent that 75% of activate swipes would be accounted for with an up/up-right/right categorisation and 79% of deactivate swipes with a down/down-left/left split.

The gesture dimensions of swipe length and speed provide new opportunities for coding for designers, and the results of this study revealed several task-related effects on how drivers make a swipe. For instance, swipes for activate/deactivate tasks were generally shorter than swipes for other task-types (see Figure 5), possibly because these activities are associated with tapping. Moreover, swipes for decrease tasks (e.g. make temperature cooler, slow down fan speed) were generally conducted at a slower speed than swipes for other tasks.

However, these results should be interpreted with caution. Further analyses revealed that gesture length and speed were both related to gesture direction (section 3.4 and Table 3). Essentially, up/down gestures were generally shorter than left/right gestures. Moreover, downwards swipe gestures were typically made more slowly than swipe gestures in other directions. Therefore, it is possible that the effects of command type on gesture length and speed were mediated by directional preferences. For example, 'deactivate' and 'decrease' gestures tended to be slower, but this might be because a large proportion of these were downwards swipes. It is difficult to confirm whether the task-related differences in gesture length and speed are due to cognitive effects (such as semantic influences of the command itself - e.g. use of phrase "slow down" for fan speed), physical effects (such as the difficulty of execution of finger movements in certain directions or the physical dimensions of the screen), or both. However, it is worth noting that increase/decrease commands prompted the largest proportion of up/down gestures, but it was the activate/deactivate commands that prompted the shortest gestures. This suggests that the effects of command type on gesture length are not entirely explained by directional differences, and that the semantics of the command are having at least some influence on swipe length.

There were few differences in the nature of swipe gestures made across the driving and static contexts. Prior to the study, we suspected that swipe direction might vary as a result of the relative touch screen location either in front (static) or to the left (driving) with respect to the participant. Specifically, we felt that the driving orientation would lead to more 'accepting/rejecting' gestures with movement towards/away from the body. Although this effect was present for activate/deactivate tasks in the driving condition, similar gesture directions were executed for the static context. Further work could explore more the possibilities of these gesture types. We did find that swipes were generally made more quickly when in a static context, as opposed to when driving. Such a difference is most likely to be a result of the divided-attention nature of driving in which fewer resources were available to the execution of the secondary (swipe) task. However, it may also have been because the majority of participants in the driving situation were using their non-preferred hand to make the swipe gesture. As a result of these factors, performance on the swipe task was likely to suffer.

A limitation of our study was the use of a fixed hotspot as the starting point for a gesture. In this initial study, it was felt to be important to control the starting point of the gesture from the centre of the screen. Nevertheless, an implemented gesture may use a much wider area of the screen, particularly to eliminate the necessity for precision movements towards buttons. With such an unconstrained starting point, it is possible that drivers would naturally start gestures closer to them (on the right side of a screen for a right-hand drive vehicle). In addition, it is worth noting that the current study did not consider the presence of a bezel surrounding the screen and the potential impact that such a tactile reference point might have on the nature of a gesture made by a driver. Indeed, it is possible that new forms of swipe gesture could be coded based on drivers starting or finishing on the bezels of invehicle touch screens.

As a final point, it is worth acknowledging the potential for a range of studies following on from our work considering how natural gestures are executed on a touchscreen in a vehicle/driving context. Examples of variables that might be explored include those related to: individual differences (handedness; prior experience with touchscreens/smartphones, etc.); task (gesture type, framing of instruction, etc.); and environment (e.g. right-hand versus left-hand drive vehicles; impacts of vibration, etc.).

5. REFERENCES

- [1] Crundall, D., Chapman, P., Trawley, S., Collins, L., Van Loon, E., Andrews, B., and Underwood, G. 2012. Some hazards are more attractive than others: Drivers of varying experience respond differently to different types of hazard. *Accident Analysis & Prevention* 45, 2, 600-609.
- [2] Good, M.D., Whiteside, J.A., Wixon, D.R., and Jones, S.J. 1984. Building a user-derived interface. *Communications of the ACM* 27, 10, 1032-1043.
- [3] Kane, S.K., Bigham, J.P., and Wobbrock, J.O. 2008. Slide Rule: Making mobile touch screens accessible to blind people using multi-touch interaction techniques. *Proc.* ASSETS '08, ACM, 73-80.
- [4] Kane, S.K., Wobbrock, J.O., and Ladner, R.E. 2011. Usable gestures for blind people: Understanding preference and performance. *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '11)*, 413-422.
- [5] Land M. F. and Horwood J. 1995. Which parts of the road guide steering? *Nature (Lond.)* 377: 339-340
- [6] Liang, Y. and Lee, J. D. 2010. Combining cognitive and visual distraction: less than the sum of its parts. *Accident Analysis & Prevention*, 42, 3, 881-890.
- Morris, M.R., Wobbrock, J.O. and Wilson, A.D. 2010. Understanding users' preferences for surface gestures. *Proc. GI '10*, Canadian Information Processing Society (2010), 261-268.

- [8] NHTSA 2006. The Impact of Driver Inattention on Crash Risk: An Analysis using the 100-car Naturalistic Driving Study, National Highway Safety Administration, Report No. DOT HS 810 594
- [9] O'Neill, E., Kaenampornpan, M., Kostakos, V., Warr, A., and Woodgate, D. 2006. Can we do without GUIs? Gesture and speech interaction with a patient information system. *Personal and Ubiquitous Computing* 10, 5, 269-283.
- [10] Ruiz, J., Li, Y., and Lank E. 2011. User-defined motion gestures for mobile interaction. *Proc. CHI 2011*, ACM, 197-206.
- [11] Sánchez, J. and Maureira, E. 2006. Subway mobility assistance tools for blind users. *Proc. ECRIM '06*, Springer (2006), 386-404.
- [12] Schuler, D. and Namioka, A. (1993). Participatory Design: Principles and Practices. Hillsdale, NJ: Lawrence Erlbaum.
- [13] Tu, H., Ren, X., and Zhai, S. 2012. A Comparative Evaluation of Finger and Pen Stroke Gestures. *Proceedings*

of the ACM Conference on Human Factors in Computing Systems (CHI 2012, 5- 10 May 2012, Austin, Texas), ACM Press (ISTP, ACM), 1287-1296

- [14] Wobbrock, J.O., Aung, H.H., Rothrock, B., and Myers, B.A. 2005. Maximizing the guessability of symbolic input. *Ext. Abstracts CHI* '05. New York: ACM Press, 1869-1872.
- [15] Wobbrock, J.O., Morris, M.R., and Wilson, A.D. 2009. Userdefined gestures for surface computing. *Proc. CHI '09*, ACM (2009), 1083-1092.
- [16] Wu, M., Shen, C., Ryall, K., Forlines, C. and Balakrishnan, R. 2006. Gesture registration, relaxation, and reuse for multipoint direct-touch surfaces. *Proc. TableTop '06*. Washington, D.C.: IEEE Computer Society, 185-192.
- [17] Zhao, S., Dragicevic, P., Chignell, M., Balakrishnan, R., and Baudisch, P. 2007. EarPod: Eyes-free menu selection using touch input and reactive audio feedback. *Proc. CHI* '07, ACM, 1395-1404.