

Direct Manipulation in Tactile Displays

Aakar Gupta, Thomas Pietrzak, Nicolas Roussel, Ravin Balakrishnan

► **To cite this version:**

Aakar Gupta, Thomas Pietrzak, Nicolas Roussel, Ravin Balakrishnan. Direct Manipulation in Tactile Displays. CHI 2016 - Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, May 2016, San Jose, United States. pp.3683 - 3693, 2016, <10.1145/2858036.2858161>. <hal-01417855>

HAL Id: hal-01417855

<https://hal.inria.fr/hal-01417855>

Submitted on 16 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Direct Manipulation in Tactile Displays

Aakar Gupta¹, Thomas Pietrzak², Nicolas Roussel³, Ravin Balakrishnan¹

¹University of Toronto, Canada, ²Univ. Lille, France, ³Inria, France

aakar@dgp.toronto.edu, thomas.pietrzak@univ-lille1.fr, nicolas.roussel@inria.fr, ravin@dgp.toronto.edu

ABSTRACT

Tactile displays have predominantly been used for information transfer using patterns or as assistive feedback for interactions. With recent advances in hardware for conveying increasingly rich tactile information that mirrors visual information, and the increasing viability of wearables that remain in constant contact with the skin, there is a compelling argument for exploring tactile interactions as rich as visual displays. Direct Manipulation underlies much of the advances in visual interactions. In this work, we introduce the concept of a Direct Manipulation-enabled Tactile display (DMT). We define the concepts of a tactile screen, tactile pixel, tactile pointer, and tactile target which enable tactile pointing, selection and drag & drop. We build a proof of concept tactile display and study its precision limits. We further develop a performance model for DMTs based on a tactile target acquisition study. Finally, we study user performance in a real-world DMT menu application. The results show that users are able to use the application with relative ease and speed.

Author Keywords

Tactile displays; Direct Manipulation; Wrist; Wearables

ACM Classification Keywords

H.5.2 Information interfaces and presentation: User Interfaces - Haptic I/O

INTRODUCTION

Engaging with visual displays is not always feasible or desired. They could be absent in certain contexts due to their fragility, cost or high battery consumption. Visual attention could be engaged elsewhere in an activity or be impaired due to disabilities. With the rise of wearables that remain in constant, sturdy contact with the skin, and the advances in tactile display capabilities, it is imperative for HCI to research tactile interactions that are as independent and as advanced as visual interactions.

The concept of direct manipulation [21] underlies much of these advances in visual interactions in the past decades. Direct manipulation is an interaction system which represents task objects on a display (visual, in general) and allows the

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 the author(s) 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.

CHI '16, May 07 - 12, 2016, San Jose, CA, USA

Copyright is held by the owner/author(s). Publication rights licensed to ACM.

ACM 978-1-4503-3362-7/16/05...\$15.00

DOI: <http://dx.doi.org/10.1145/2858036.2858161>

user to directly interact with them through actions whose effects are also continuously displayed. This continuous cycle of control and progress forms the core of direct manipulation.

The notion of direct manipulation in this form is absent in current tactile displays. A great deal of work in digital tactile displays is on tactile pattern recognition and feedback for interactions. In fact, sensory substitution techniques have successfully delivered tactile patterns akin to visual or auditory information [14, 13]. However, in almost all cases, tactile displays function as feedback or pattern notification tools and not as displays that enable direct manipulation of *tactile objects* in a *Tactile User Interface*. The advances in actuation technology and understanding of tactile perception encourage tactile interactions analogous to visual interactions.

In this work, we introduce the concept of direct manipulation for tactile displays. We describe the constructs of a *tactile screen*, *tactile pixels*, *tactile pointer*, and *tactile targets*. We describe a tactile indirect pointing interface that enables user actions like pointing, selecting, and drag & drop on tactile targets. Based on this conceptual framework, we build a tactile display for the wrist that demonstrates the feasibility of the concept. We establish the precision limits of this display via a target distinction study. We then construct a performance model for tactile target acquisition along the lines of Fitts' law. Finally, we investigate the use of a tactile-only menu application on the tactile display. The results show that the users are able to use the tactile indirect pointing interface and select, drag and drop tactile menu items with relative ease.

DIRECT MANIPULATION FOR TACTILE DISPLAYS

Imagine a pointer controlled by a regular computer mouse but which the user can *feel* moving around her wrist instead of seeing it moving on a screen. At a particular location on her wrist, she feels a change that indicates a target. She performs a double click, which executes an associated command. This is the tactile direct manipulation experience which enables the loop of *control* and *progress* for tactile displays attached to the skin. In this section we develop the concept of direct manipulation for tactile displays by deconstructing the components that make direct manipulation in visual displays work end-to-end and adapting each of them for tactile displays. We illustrate the concept using the example of a wrist-mounted circular tactile display which eventually ties into our proof of concept (POC). However, the concept applies independently to any potential tactile display on any region of the skin.

There are two basic components of a visual direct manipulation system - *display* and *input*. Display refers to a combination of a *screen* and an *interface*. Our focus in this work is on

Direct Manipulation-enabled Tactile displays (DMTs) for the body's tactile sense. Analogous to a visual screen, we define a *tactile screen* as a cluster of *tactile pixels* that enable tactile perception of system responses. A tactile screen consists of tactile objects that are combinations of tactile pixels.

An interface combines these pixels into objects that users can individually perceive. It then defines user actions to interact with these objects. Before we get to the tactile interface, we need to understand tactile perception first. With visual displays, the user is always aware of the overall state of the screen via a simple glance at the screen. This is fundamentally different from tactile perception. If multiple tactile objects in close proximity on skin get simultaneously stimulated to make the user aware of the tactile screen's overall state, user feels a single composite sensation and is not able to discern individual tactile objects accurately [22]. The tactile sense requires active exploration to identify objects and their forms [9]. Consequently, users need a way to explore or track objects on the tactile screen without performing actions on them.

This leads us to Indirect Pointing Interfaces where the input space is different from the display space. A tracking state where the system is aware of the tracking by the user and responds accordingly is an inherent property of indirect pointing interfaces as described in Buxton's 3-state model [3]. In direct touch interfaces there is typically no tracking state (unless an additional modality like pressure is used) and interaction with the screen results in an action on the objects. In indirect pointing, the virtual pointer tracks objects without performing actions on them. The pointer is perfect as the virtual tool using which a user can explore the tactile interface. Thus, we choose indirect pointing interfaces as the interface paradigm for DMTs. A tactile indirect pointing interface will analogously have a *tactile pointer* that allows tracking of the tactile screen, and *tactile targets* that are tactile objects that the user can act upon. The user actions will analogously be tactile pointing, selection & execution, and drag & drop.

We formalize the DMT by defining the tactile screen and tactile indirect pointing interface and their respective components. For the *input*, we illustrate the concepts using a touchpad which we use in our POC as well. However, any indirect pointing device can potentially be used for DMTs.

The Tactile Screen

Analogous to a visual screen, the tactile screen has three defining attributes - *shape* (or geometry), *size*, and *resolution*. The tactile screen is composed of a cluster of individual pixels, termed *tactile pixels*, each of which have three defining attributes - *coordinates*, *stimulus pattern range*, and *stimulus strength range*.

For a tactile device worn or affixed on the skin, the tactile *shape* and *size* of the screen are defined by the region on the skin within which the device can convey tactile sensations. In contrast to visual screens, this could depend both on the device hardware, as well as the anatomy of the body part being instrumented. For example, for a stretchable wrist band consisting of tactile actuators, the exact shape depends

on wrist anatomy, and size depends on circumference around the wrist. However, for a tactile device to be used across multiple users with the same interface, these should have fixed values. An abstraction is required which fixes the effective shape and size for a particular device. For our POC, we fix it to a circular screen comprising of 360 degrees. The absolute measure of each degree would be different for every wrist, but the abstraction allows a developer to make apps which can work within a certain range of wrist sizes.

Before we come to tactile resolution, we define a tactile pixel. A *tactile pixel* is the smallest tactile sensation that the device can distinctly modify on the tactile screen. This distinction might or might not be perceivable by the user, depending on the pixel size (as is the case with visual displays). The modifiable attributes are stimulus pattern & stimulus strength. *Tactile resolution* is the total number of tactile pixels on the screen. In our POC, we use phantom sensations such that four actuators around the wrist enable 360 tactile pixels, one for each degree, making its resolution 360x1. It is a one-dimensional (1D) tactile screen where the pixels are spread on the wrist circumference. Theoretically, in a very simple tactile display, an individual actuator could be the tactile pixel.

Every tactile pixel has a location *coordinate* depending on where it is on the screen. For the POC wristband, coordinates span from 0° to 360°. The *stimulus pattern* and *strength* attributes are analogous to the visual pixel color and brightness. Both of them should be similarly distinguishable for certain discrimination thresholds within the stimulus's range. This would depend on the tactile actuation. For vibrotactile actuators, frequency and amplitude have been shown to have discrimination thresholds within a certain range and therefore the POC uses them for pattern and strength respectively.

The Tactile Indirect Pointing Interface

The indirect pointing interface consists of a tactile pointer and tactile targets. Interaction with tactile targets happens via tactile pointing, selection, execution, and drag & drop.

Tactile Pointer

The tactile pointer enables the user to navigate the tactile screen. To avoid conflicting sensations, the device's tactile sensation is stimulated only at the location coordinate of the pointer. We refer to this as the tactile response of the DMT. Initially, the pointer is at a certain coordinate on the screen. As the user controls the pointer using an input device, the pointer starts moving from its current location to the neighboring pixels until the user stops. The user feels this pointer movement as a tactile sensation going from one skin location to another depending on the speed of user's control movement. In the ideal case, the user feels a precise point-like sensation going smoothly from one location on skin to another.

The specific tactile sensation felt by the user at the pointer location depends on the tactile object that is at the current pointer location. Tactile pointer is only the virtual tool for navigating the screen and does not have a tactile response of its own. There are only two types of tactile objects: voids and targets. Targets are objects that the user can act upon. In places where there is no target, there is a void.

Tactile Targets & Voids

The tactile interface relies on users differentiating between i) a void from a target, and ii) a target from another target. While voids are all same, targets can be different from each other. Tactile response of a void when the pointer is at its location should be perceived differently from that of a target. To achieve this, we dig into the attributes of a tactile object.

A tactile object is a combination of multiple pixels with the following properties - *size*, *location*, and *pixel attributes* of all pixels in the object. Size and location are derived attributes from the pixel attributes of all pixels since each of them have a coordinate. In a 2D tactile screen, the objects do not necessarily have to be rectangular, which means that aside from size and location, shape would also be a derived attribute. To keep things simple, that case is not addressed. Since voids and targets are next to each other, they need to be distinguished based on perceivable properties of the pixel like pattern or strength. Voids are assigned a low strength or pattern stimulus distinguishable from higher values reserved for targets. For the POC, we use a fixed lower frequency for every pixel on the void to signify when the pointer is over it, and higher frequency for the targets.

The frequency and amplitude discrimination is relative. Since targets are explored one by one, the variety of different targets needs to be limited on the tactile screen or else the user will lose track of the relative distinction. For the POC, we use the simplest case where only the *location* property is used to distinguish between multiple targets. All targets have the same amplitude and frequency over all of their pixels. Figure 1 illustrates the void and target discrimination. In figure 1(a), the pointer is over a void where the user can feel the void's low frequency tactile response. The user then moves the pointer on the screen, all the while feeling the void tactile response following the pointer movement. In figure 1(b) when the pointer reaches a target, the user feels the target's high frequency tactile response. Going further, the user will encounter a void and then a second target, which the user can discriminate using localization.

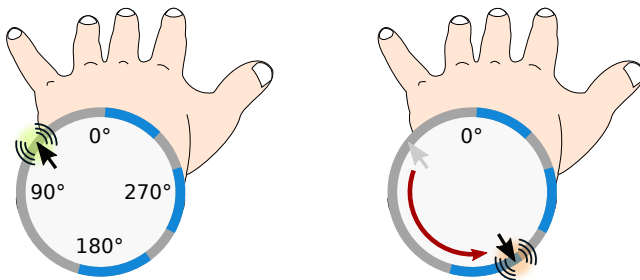


Figure 1: A 1D circular 360x1 tactile display around the wrist. On left, tactile pointer is over a void whose tactile response frequency is represented in green. On right, user navigates the pointer to a target, where the tactile response frequency is different (orange).

However, the exploratory movement of the pointer on the screen adds another dimension to target discrimination besides its location. Unlike localization, where the user needs to localize a random tactile sensation, here the user starts moving from a certain location on the screen to eventually reach the desired target location and therefore has an additional temporal sense of how far the pointer has traveled. Further,

users can device mechanisms such as maintaining a count of their favorite targets from a fixed location and simply count the targets to reach the desired one. In fact, users can use location, time, and counting in conjunction to guide their movements. We term this exploratory usage of finding targets that are differentiated only by location on a DMT as *exploratory localization*.

Tactile Indirect Pointing Interface Actions

Before describing user actions on tactile objects using tactile pointer, we describe the input used for illustration: a multi-touch screen used as a touchpad with no visuals. A finger down (FD) and move (FM) is equivalent to the mouse hover motion. The second finger tap while first is down and then lift up of both fingers (SFT) is the mouse left click. The second finger moving (SFM) while the first finger is down and static is mouse move with left button down. A second finger double tap (SFDT) while the first finger is down and then lift up of both fingers is mouse double click. We could have used any input device, including the mouse, but this helps maintain consistency with our POC.

Each action description is supported by a state transition diagram. Every state is a unique state of the tactile pointer: over void (OV), over target (OT), over selected target (ST), dragging over void (DV), dragging over target (DT). The color of state bubbles in the figures refer to the mode: Idle (Blue), Tracking (Light Green), Dragging (Yellow). Every state has a unique tactile response, which is shown by a green R_x label. The input action causing the transition is shown by a black label. When an action results in a command (Select, Execute etc.), the command is shown with a red label. Every command can either be a success or failure. This is conveyed to the user via short instant pulses of tactile responses at pointer location. These are different from state tactile responses and are termed as *ephemeral tactile responses*. Each command's success and failure has a unique ephemeral response. The ephemeral responses are played during a short pause after the tactile response of the current state stops and the next state starts.

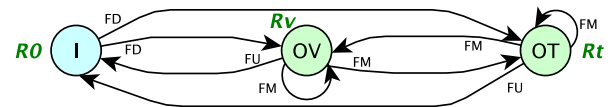


Figure 2: State Transitions for Pointing

Pointing

When no finger is on the screen, the pointer is idle (I - idle), and the tactile response is R_0 which should be zero in normal instances. When the user puts the finger down, she feels the tactile response R_v of void (OV) or R_t of target (OT) depending on what's under the current pointer location. As the finger moves (FM), the pointer moves correspondingly over voids and targets and the user feels R_v/R_t depending on the state. The size of a target can be small or large and the pointer can keep moving over the same target for some time if the pointer is moving slowly. In our POC, we enable the distinction between R_v and R_t using frequency, but other attributes can also be used. On finger up (FU), the pointer is Idle again, but still maintains its last position.

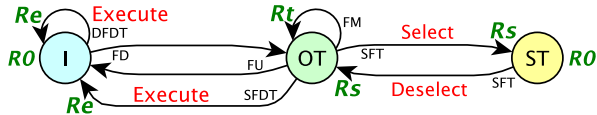


Figure 3: State Transitions for Target Execution & Selection

Target Execution and Selection

When over a target (OT), the user executes it by doing a double tap with the second finger and lifting up both fingers (SFDT). An alternate way to execute a target under the pointer while in idle state without going to the intermediate OT state is to do a double finger double tap (DFDT). Execution is a command action which results in the ephemeral response R_e . R_e has two possible values, one each for success and failure. The tactile response after execution is Idle because no fingers are on screen. The ensuing responses will depend on what happens after target execution. For instance, a target can be an icon to an application that takes over the entire screen or it could simply be a button that performs a task (like volume up) but the screen remains the same.

Similarly, after a finger down (FD), a second finger tap and lift off of both fingers (SFT) when pointer is in OT selects the target (ST). Doing the same action in ST, deselects the target. The alternate way while in idle state is a double finger single tap (DFST) (not depicted in figure). The select/deselect command results in the ephemeral response R_s . Again, R_s has two possible success and failure values. It is not necessary to keep R_s different from R_e since the user knows the action she performed and perceives success or failure for that action. Tactile response in ST is R_0 because the fingers are up immediately after selection.

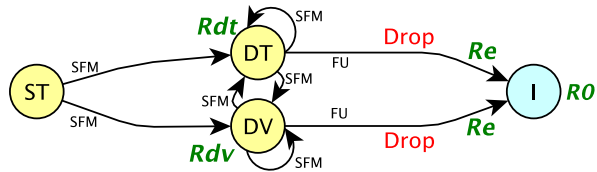


Figure 4: State Transitions for Dragging

Target Manipulation: Dragging

When a target is selected (ST), the user drags the target by moving the second finger (SFM). The pointer can drag the target over void (DV) with response R_{dv} , or over another target (DT) with response R_{dt} . Dragging is a different mode of pointer movement than tracking, and therefore the responses in DV and DT are R_{dv} and R_{dt} which can be different from R_v and R_t , but don't necessarily have to. The user eventually drops the target by lifting both fingers up (FU) at the desired location. The drop command results in the ephemeral response R_d which again has success and failure values.

We see that the tactile response of the display depends on three variables: whether pointer is over a void or target (R_v , R_t), whether the pointer is in tracking mode or dragging mode (R_{dv} , R_{dt}), and whether a command was invoked (R_e , R_s). The first two are continuous responses as long as the pointer is in that state, while the third is a short-term response signal. While our focus is on the tactile responses of each state and action, the state transitions with inputs help us understand the

end-to-end interaction. These transitions can change depending on the input mechanism used. A complete state machine of a DMT can be constructed by combining the three figures. We now have the basic building blocks of an indirect pointing interface which can be used to create higher level widgets such as lists, menus, sliders, etc.

Control & Progress in Tactile Displays

As described, while the adaptation of direct manipulation for the tactile context requires delving into the specific tactile affordances, the theoretical construct of DM remains the same. Looking at direct manipulation through a tactile lens, *Control & Progress* are two salient properties that distinguish DMTs from other existing works in tactile space. *Control* refers to the user being in control of the objects on the tactile display which governs the tactile responses that the user feels. *Progress* refers to the ability of the tactile display to update continuously in real time, in line with the user's actions, informing about her progress. In effect, the user engages in a cycle of control and progress so that later input actions depend on earlier output responses and vice versa. User controls the pointer whose progress is continuously being conveyed tactically to the user, which, in turn, guides the user's further navigation of the pointer.

DMT IN CURRENT TACTILE SPACE

Technically, *Tactile* refers to sensory stimulations that affect the sense of touch and *haptic* refers to both *tactile & kinesthetic* stimulation. Our work is focused on tactile sense. To contextualize our work better, we classify tactile interactions into four groups—pattern notifications, assistive feedback, dynamic guidance, and exploratory tactile displays. For brevity, we mention 1-2 recent works that are indicative of the group.

Pattern notifications are when different spatial and/or temporal tactile patterns are used to convey different information to users. Typically, this involves either short-term alerts [18], or long-term information like progress bars [1] or direction [11].

Assistive feedback is when the user is aided in an interaction by tactile feedback, such as eyes-free gestures [12], braille-reading [17] or color identification [4].

Dynamic guidance is when guidance information is tactically conveyed to the user. Such systems dynamically change the tactile response based on the user's movements to guide them better for purposes like motor learning [20]. The user, here, is not interacting with the computer, only getting guided.

Exploratory tactile displays require the user to explore them to convey tactile information; a braille book, for instance. Such displays mostly work when users use their fingers [23] to explore on the display.

None of the above involve the user controlling and performing actions on the tactile display while the display continuously informs on the progress of user actions in real time. In a very simplified sense, however, DMTs can be thought of as exploratory tactile displays that can be instrumented on body-parts other than the finger, and that separate the body-part doing the exploration from the body-part that feels the tactile response.

PROOF OF CONCEPT

To demonstrate that DMTs can be functional, usable systems, we build and study a prototype 1D DMT display around the wrist. With the popularization of wrist wearables that remain in constant, sturdy skin contact, the wrist presents a fitting space for tactile interventions. Prior studies show that wrist regions support rich tactile stimulation [15]. We use vibrotactile stimulation for its proven phantom sensation capabilities [10]. We now describe how the tactile screen, pointer, targets, and actions are realized in our POC. In indirect pointing, movements in input space are mapped to display space movements via a transfer function, which is described later.

Tactile Screen

A circular 1D tactile screen is apt for the wrist circumference and efficiently demonstrates the concept. Assuming the wrist to be a crude circle, the screen size is set to 360 degrees. We use only a subset of values within these ranges. The hardware consisted of four EAI C2 actuators, called *tactors*, attached to an elastic sports wristband, which has an Arduino, Bluetooth, battery, and a signal amplification board (Figure 5). The actuators were placed at top (0°), left (90°), bottom (180°), and right (270°) positions on the wrist. The device allowed tactor amplitude, frequency and duration control via Bluetooth commands. We paired it with an LG G Android watch on the wristband whose touchscreen was the input touchpad. Since the tactile display is 1D, the touchpad registered finger motion in only one dimension parallel to wrist circumference (Figure 5). The pointer moved in the direction corresponding to the direction of finger movement. Looking at Figure 5, if the finger moved in the thumb-to-little-finger direction on the watch touchpad, the pointer moved in the 0° to 90° direction, and similarly reverse for reverse finger movement. During tactile interactions, the touchpad showed no visuals. During experiments, the touchscreen was used between trials for the logistics of the experiment. The tactor frequency range is 0-400Hz. The C2 tactor prescribes an amplitude range between 0-255. We only use the maximum amplitude and a select few frequencies for our DMT.

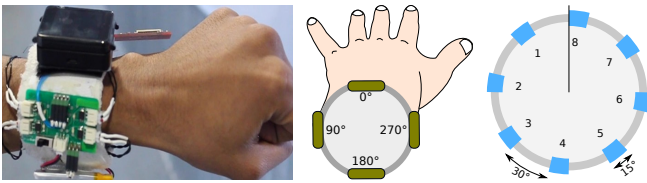


Figure 5: a) Wristband, actuator positions b) Study 1 sample layout

Tactile Pointer Implementation

We simulate a tactile pointer whose movement can be controlled over every pixel and felt *continuously* on the skin. The continuity is crucial to the feeling of smooth pointer movement and something like an array of mini actuators would not be ideal. We utilize the funneling sensory illusion [22]: the simultaneous stimulation of two tactile actuators in close proximity on skin results in a single illusory stimulation at the location between the real actuators. These are called phantom sensations. While phantom sensations have been extensively explored to generate localized stimulation, their use for stimulating continuous tactile movement has been limited to two

actuators [19] which simply generate a quick pulse in a particular direction. They do not go into the details of movement speed and precision, as well as, of giving the control of the sensation to the user.

The aim of the tactile pointer is to traverse over every pixel with a speed controlled by the user. To stimulate a tactile response at the 45° tactile pixel, the actuators at 0° and 90° are played at equal intensities. A tactile pixel at θ° is stimulated by varying the amplitude of the two actuators between which θ° lies. We adapt Israr *et al.*'s point stimulation algorithm [10] for our screen to get the amplitudes:

$$A_1(\theta) = \pm A \sin(\theta) \quad A_2(\theta) = \pm A \cos(\theta)$$

where A_1 corresponds to the actuator at the lower angle among the two. A is the effective amplitude of the stimulated pixel which is preset and follows the formula: $A = \sqrt{(A_1(\theta)^2 + A_2(\theta)^2)}$. We fix A at a C2 maximum of 255 for all vibrations. Stimulating adjacent pixels in rapid succession based on user control produces the sensation of the tactile pointer moving around the wrist.

Tactile Target Implementation

To implement tactile targets, we designate a different actuator vibration frequency for the target and the void. The user tracks the pointer over voids (OV) using phantom vibrations playing at a frequency $F_v = 75\text{Hz}$ for C2 tactors. When the pointer is over a target (OT), the vibrations switch to $F_t = 320\text{Hz}$ which is perceived differently from F_v .

Tactile Interface Actions Implementation

The difference in tactile responses depends on frequency. R_v & R_{dv} have the same frequency F_v and R_t & R_{dt} have F_t . The ephemeral responses R_e , R_s , and R_d all have the same success response, and same failure response. Success is one quick pulse and failure is two quick pulses. Success response sequence: 100ms pause, 300ms pulse at 320Hz, 100ms pause. Failure response sequence: 100ms pause, 100ms pulse at 320Hz, 100ms pause, 100ms pulse at 320Hz, 100ms pause.

Transfer Function

Our transfer function maps the watch touchpad pixels to display degrees. The scale factor between movement velocities in the two spaces is termed as CD gain = $V_{display}/V_{input}$ [5]. We use a discrete logistic function approximation as this gain:

$$gain = \begin{cases} g_l & \text{if } v_{input} < v_l(\text{mm/s}) \\ g_u & \text{if } v_{input} > v_u(\text{mm/s}) \\ g_u + (g_u - g_l) \times \frac{v_{input} - v_l}{v_u - v_l} & \text{else} \end{cases}$$

Distance moved by the tactile pointer is: $d_{display}(\text{degrees}) = gain \times d_{input}(\text{mm})$. $g_l = 0.5$, $g_u = 1.29$, $v_l = 113\text{px/s}$, $v_u = 10,000\text{px/s}$ were fixed after a pilot. All values in the implementation were arrived at after a series of pilots with four users.

While localization has been studied on the wrist [16, 6], DMTs use *exploratory localization* which potentially enables users to work with more targets. We study how precisely the DMT allows users to move and perceive the pointer and make a distinction between targets. We then study target acquisition in DMTs and formulate a DMT performance model. Finally, we study a DMT menu application and see how users perform and use the DMT.

STUDY I: MOVEMENT & TARGET DISTINCTION

Phantom sensations and frequency discrimination worked well during our pilot studies. However, the limits of movement precision and accurate target discrimination that these enable in a DMT need to be studied. We conducted a study where participants were asked to count a random number of targets on screen. This required participants to control speed of movement depending on number of targets present. If the system allows precise movement, accurate perception, and precise target discrimination, participants should be able to count large numbers of targets accurately.

The participants did multiple trials. The initial number of targets to be counted was 4. For the next trial, the count was exponentially increased ($\text{round}(4 \times \text{targets} \times 10^{0.5}) = 13$ targets) or decreased based on if the answer was correct or not. A change from increasing to decreasing targets, and vice versa, is referred to as a *reversal*. After first 3 reversals, the exponential factor was reduced from 0.5 to 0.1. As target count increased, the corresponding target size and distance between targets decreased, thus requiring the user to be more controlled in their movement to perceive targets differently from the void and from other targets. The large initial factor ensured faster convergence of the target count to the participant's performance threshold. The smaller value later ensured fine resolution of the threshold estimate. The experiment was terminated after 8 reversals at the smaller value. The average number of targets from these 8 reversals was taken as an estimate of the threshold level. This threshold represented the upper limit of target count which can be usefully put on the tactile screen.

Equal-sized targets were uniformly distributed on screen. The ratio between target and void size was kept constant at 1:2. So, as the count increased, target size decreased. The pointer started at the origin at 0° . Figure 5-b shows the 8-target layout. The experiment ran in an app on the smartwatch. Since the task only involved pointing and not target selection, a double finger tap was used to stop a trial which led to a visual screen on the touchscreen for the user to enter the answer for this trial. On pressing *Ok* on this screen, the blank touchpad appeared again and next trial began.

7 participants (6 male, mean age = 25.6 years) with wrist sizes within 150-178mm took part. Participants were recruited using flyers in a University campus. Participants wore the wristband in left hand, which is usually the watch hand. They also wore headphones playing pink noise to mask tactor sounds. Each session lasted 15-25 minutes. The session started with a system demo, followed by the task. The instruction was "give an accurate count and try to be as quick as possible". The experiment began after a dummy trial with 2 targets.

Results

Mean threshold count was 19.1 ($SD = 7.2$). This is lower than the mean of the maximum counts that the participants got right - 29.6. Essentially, for a normal wrist of 160mm, a target at every 8mm is recognizable in normal use which is quite precise. Therefore, phantom sensations were found to work well with frequency discrimination to enable precise movement and target discrimination on the DMT. No correlation was found for wrist size.

STUDY II: A PERFORMANCE MODEL FOR DMTS

Reliable prediction models of user performance are imperative to understand device capabilities and human limits. Movement Time (MT) models for target acquisition have been studied using Fitts' law for a number of input devices where the output is a visual display. Fitts' law relates MT to target width (W) and distance (A): $MT = a + b \log_2(2A/W)$. Tactile displays are different from visual displays. They do not give an overview of the screen at a glance and are subject to variances in perception. Whether the same model and dependencies hold for tactile displays, is a nontrivial question that needs to be investigated.

A target acquisition task has a ballistic phase where the pointer moves to the general region of the target, and a corrective phase when the pointer makes small movements to reach the exact location. The ballistic phase is responsible for large distances leading to higher MT s. The corrective phase is responsible for smaller widths leading to higher MT s. We hypothesize that this general rule should apply to the tactile display - however, perception variations of different target positions could be a significant factor that affects MT and its relationship with width and distance. We conducted a target acquisition study that asked the following questions:

- How do target width (size) & distance affect MT in DMTs?
- Is there an effect of target position on MT in DMTs?
- What is the target acquisition performance? If and how does it improve over time for DMTs?

The last question informs the performance metrics of our specific wrist DMT, it will give us a sense of what to expect from such systems in general. The basic relationship between MT and width, distance, and target position that the experiment yields should hold more generally for DMTs.

Experiment Method

A point and select task was designed with the following independent variables: DISTANCE (3 levels), WIDTH (3 levels), POSITION (8 levels) and BLOCK (4 levels). 9 participants, different from previous study, (6 male, 8 right-handed, mean age=29.8) took part. The experiment lasted 20-30 minutes.

We followed the no-distractor target acquisition study design [8], where during a single trial, only one of the positions functioned as the target. There were eight fixed targets which allowed exploration of all wrist regions and kept experiment duration reasonable. Mid-points of targets were at 0° , 45° , 90° , 135° , 180° , 225° , 270° & 315° . The three width levels were selected such that the void width:target width ratios were 1,3, and 5. After rounding off, the widths were 24° , 13° and 9° respectively. For instance, a target at 90° with a width of 13° spanned 84° - 96° . The log scale separation in width levels was in line with earlier pointing studies [8]

Participants needed to go from one target to the next without reinitializing the pointer. As is standard practice [8], distance was the degrees between mid-points of the previous target and current target. The three distance levels were 45° , 90° & 180° . Since the display was circular, the minimum distance between two targets cannot exceed a maximum of 180° . The user could possibly go in either direction and figuring out the

optimal direction was part of user performance. The final evaluations used the minimum distance, no matter which direction the user took. In total, $8 \times 3 \times 3 = 72$ conditions (trials) = 1 block. We study 4 blocks of 72 trials each, with the first one intended as a training block.

Experiment Design

A within-subjects study was run for all variables. The sequence of 72 conditions in a block was such that starting with the first, each condition led to the next, while ensuring that no position, width, or distance value appeared consecutively. 36 such sequences were designed for the $9 \times 4 = 36$ blocks.

The experiment ran in a smartwatch app. The task was designed analogous to a visual target acquisition study. As soon as the participants made a selection for the current target, the ephemeral responses informed them of their success or failure. After a 1s delay, participants were informed of the next target by a vibration pulse at the target position. This utilized the participants' ability to coarsely locate a stimulation on the skin. We hypothesized that as blocks progressed and participants locate the positions repeatedly, they'll start doing quick shorthand finger movements as opposed to active exploration.

The task was preceded by a demo and a dummy trial. Participants were instructed to *work as fast as possible while still maintaining high accuracy* [8]. The first trial of every block began with the pointer at 0° . For subsequent trials the pointer position was where the participant left the pointer in the previous trial. Participants were prompted to take a break after every 24 trials, with the pointer position at 0° after the break. Participants wore headphones playing pink noise. In total, we had 9 PARTICIPANTS \times 4 BLOCKS \times 3 DISTANCES \times 3 WIDTHS \times 8 POSITIONS = 2592 measurements.

Results

For 3 blocks of trials (excluding the training block), the mean *MT* per trial was 3.36s ($SD = 0.48$). The mean accuracy was 93.7% ($SD = 9.21$). A four-way repeated measures Anova analysis found significant main effects of width, distance, and block on *MT*: block ($F(2, 16) = 7.292, p < .01, \eta_p^2 = .477$), distance ($F(2, 16) = 17.944, p < .01, \eta_p^2 = .692$), width ($F(2, 16) = 40.708, p < .01, \eta_p^2 = .836$). No interaction effects were found. We now report on participant performance and improvement, followed by the effects of width, distance and position, and finally propose a performance model.

Performance and Improvement

A 93.7% accuracy is promising for an interface completely relying on the tactile sense. The *MT*, however, of 3.36s is relatively slow. A post-hoc test with Bonferroni correction for block showed a significant difference between blocks 2 and 4 ($p < 0.05$). Figure 6-c shows that mean *MT* decreases as the blocks progress. The mean accuracy in the initial training block was low at 85.03% with a high standard deviation of 24.6. We conducted a multiple logistic regression analysis of the success output of all four blocks and found a significant overall effect of block on accuracy ($Wald = 51.007, p < .01$). Figure 6-f shows a sharp increase in accuracy after the first block, followed by a relative stabilizing of the accuracy. Both

speed and accuracy curves show that despite initial friction, users will get better with practice on the DMT.

Effects of width and distance

The post-hoc tests showed a significant difference between *MTs* of all 3 width pairs: 9° & 13° ($p < .05$), 13° & 24° ($p < .01$), 9° & 24° ($p < .01$). Figure 6-a shows the mean *MTs* decreasing as the width increases. The 9° target translates to a miniscule target of 4mm for an average 160mm wrist. *MT* reduces by an entire second from a 9° target to 24° . Thus, limiting minimum target widths on DMTs is recommended.

We ran a multiple logistic regression to model binary success output based on the four variables. A significant effect of width on accuracy was found ($Wald = 14.547, p < .01, \beta_{width} = .065, OR = 1.067$). Figure 6-e shows accuracy increasing by width. Even for a target as small as 4mm, accuracy is at a healthy 91.5%. Aside from higher corrective adjustments, acquiring the smallest targets was challenging because of constant clutching. Participants found it difficult to gauge if the pointer slightly slid off the target when lifting the finger. This issue is addressed in the next study.

For distance, the post-hoc tests showed a significant difference between the *MTs* of 45° & 180° ($p < 0.01$), and 90° & 180° ($p < .01$). Figure 6-b shows that the mean *MT* increases with distance. For 45° & 90° , a 45° difference is not enough to significantly impact *MT* due to the fast ballistic movement. Participants took the optimal direction for minimum distance in 93.7% trials. (This excludes the symmetric 180° distances.) Even without an overview of the display, participants knew the shortest path direction. This demonstrates their use of *exploratory localization* where they select direction based on target location and then explore locally to find the target. The effects of width & distance confirm our hypotheses. The respective inverse and direct relation with *MT* hold true for tactile displays.

Effects of Position

There was no position effect in the initial analysis. However, participants noted the perception being better in certain regions. We re-ran the analysis, this time grouping the positions into categories. A significant effect of position on *MT* was found for the categorization: Anchor ($0^\circ, 90^\circ, 180^\circ, 270^\circ$) & Intermediate ($45^\circ, 135^\circ, 225^\circ, 315^\circ$) ($F(2, 8) = 7.608, p < 0.05, \eta_p^2 = .487$). Figure 6-d shows the anchor positions had a lower *MT* than intermediates. This has two possible explanations. First, the actuators were placed at anchors. Second, the wrist top, bottom, sides are natural anchors on the skin which a user uses to gauge relative pointer position. This increases tactile acuity due to the anchor effect [7]. Looking at the difference between means and effect sizes (η_p^2), the effect of width & distance is stronger than position. Therefore, while position should be a design consideration for DMTs, it does not supersede width & distance. The assertion, of course, is based on the wrist DMT and generalizations should be carefully considered.

Performance Model

We fit a variant of Fitts' law for *MT* in DMTs. A multiple linear regression was run for *MT* based on $\log_2(2A/W)$,

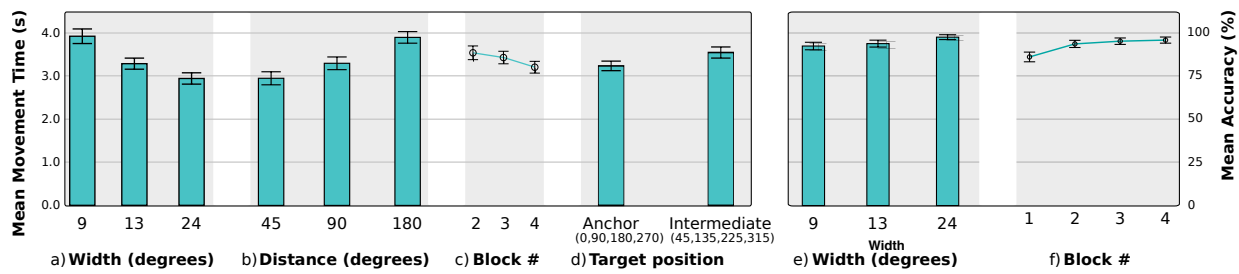


Figure 6: Influence of target width, distance, block and position on pointing time; and width and block on accuracy.

the combined width-distance variable from Fitts' equation, and P , a nominal variable with values- Anchor/Intermediate. To model end performance, the test was run on the last block. It gave the following equation for MT ($F(2,646) = 1196.781$, $p < .001$, $R^2(adjusted) = .787$):

$$MT = b_1 \log_2(2A/W) + b_2 P$$

Here $b_1 = .802$, $b_2 = .120$. Both variables added significantly to the prediction, $p < .001$. The model adheres to Fitts' law with an additional position term. These values are obviously affected by the choice of touchpad as input. It is a reasonable claim that DMTs will generally follow the width & distance relationships with movement time while adhering to Fitts' law. Position's effect is less impactful, but is subject to the body part and actuation and should not be discarded.

STUDY III: A TACTILE MENU APPLICATION

We have investigated the limits of tactile target distinction and established a performance model for DMTs. These show that DMTs are capable and usable for direct manipulation. The next question is *how well can the users perform in an actual application?* The question also pertains to how users get started with such a tactile interface and can this tactile interface be used independently without a visual display. To investigate this, we study a DMT menu application where the users perform tactile target execution and drag & drop. The study starts with a static visual aid (Figure 8) to understand the visual to tactile mapping. The aid is subsequently removed to study independent DMT use. Eight participants, different from previous studies, (all male, mean age=26.5) took part. The experiment lasted 75 mins. Participants were debriefed following the study and were asked to rate the physical and mental demand of the tasks on a five-point likert scale.

Design

The menu application is modeled after a frequent contacts list. Two menu sizes were studied: 4 items (targets) and 8 items (Figure 8). It was divided into 4 stages in the following order: item execution for 4 items, item execution for 8 items, drag & drop for 4 items, and drag & drop for 8 items. This progression of stages enabled the users to carry their learning over from previous stage to the next.

For each stage in item execution, users performed 15 blocks. With 4 items, each block consisted of 3 execution trials, with *Bob*, *Dan*, and *Helen* randomly distributed in each. With 8 items, each block consisted of 4 trials, with *Bob*, *Dan*, *Emma*, and *Gary* randomly distributed. The 15 blocks were divided in 3 sessions, consisting of 5 blocks each: In blocks 1-5, the

users could see a static visual map of the menu on a desktop (Figure 8). In blocks 6-10, the map was hidden, and was shown only upon request in a trial. In blocks 11-15, the visual map was not shown at all. Each individual session took 1-3 minutes. As evident, users were presented with only 3 out of 4 items, and 4 of 8 items. Limiting items follows the idea that not all menu items are accessed equally and some are rarely used. The screen still had 4 or 8 items depending on the stage, all of which the users felt during a trial. Item size was fixed at 11° . Before each trial, users were shown the name of the item to be executed on the watch. Pressing "Ok" on the touchscreen started the blank touchpad. The tactile pointer started at 22° for every trial. Upon the execution action, ephemeral responses informed users of their success or failure.

The drag & drop task was dragging a source item to a destination item. For each stage in drag & drop, users performed 10 blocks of 2 trials each. With 4 items, the two trials had mutually exclusive random combinations of source & destination items, thus involving all 4 items. It was similar for 8 items, with 4 items *Bob*, *Dan*, *Emma*, and *Gary*. Since, users were familiar with the menu layouts after the execution tasks, the map was shown only upon request for a trial in blocks 1-5, no visual map was allowed in blocks 6-10. Each individual session lasted 2-4 minutes. Before each trial, users were shown a text instruction, for example, "Drag *Bob* to *Dan*".

Task Input

Based on user feedback, we modified the input mechanism for efficiency. Finger movement in earlier studies was along the single touchpad dimension parallel to wrist circumference. However, as mentioned earlier, constant clutching was a problem. To remove clutching, input was modified to a circular path on the touchpad, close to the bezel, like an ipod wheel, where the pointer movement corresponded to the degrees traveled on the circular path. A constant CD gain of 0.5 was fixed. A 2° movement on touchpad led to a 1° movement on the tactile screen. The input actions for drag & drop were changed accordingly: user selects the source item using a double finger single tap, then moves it with a single finger down & move. The item is dropped when the user lifts the finger up.

A visual menu does not have spaces between different items. However, our DMT needs the void between two items for item distinction. To solve this, an activation zone was designated for each target, so that users could perform actions (execute/select/drop) for that item within that item's zone even when the item was felt only in its 11° span. This activation zone is the arc within the dotted lines around an item (Figure

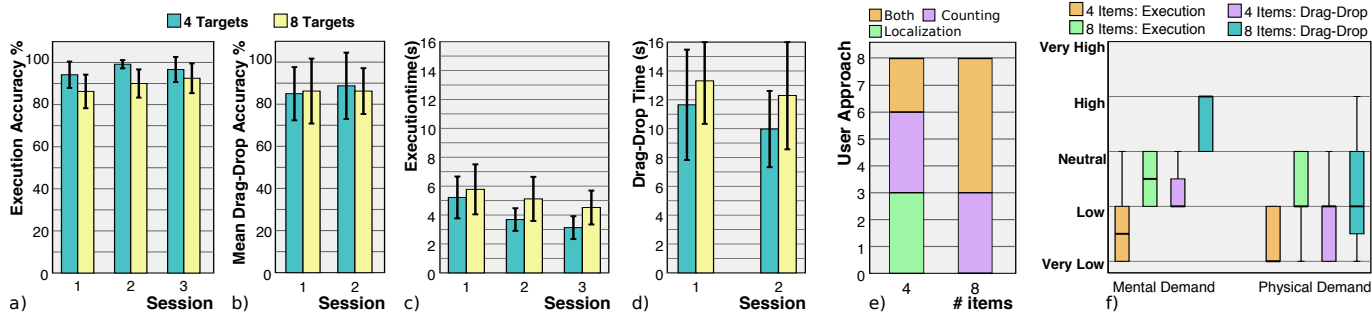


Figure 7: Study 3 Menu Application Results: (a-d) Mean Accuracy and Time for Execution and Drag & Drop tasks. (e) User approach to finding an item. (f) Mental and Physical Demand of the tasks.

8). This ensured a menu-like functionality while keeping the item distinction intact. This also addresses the problem of incorrect answers when the user is only off by a few degrees. The experiment procedure was similar to the previous study. For 8 PARTICIPANTS, for each stage (BLOCKS x TRIALS) = $8 \times (15 \times 3 + 15 \times 4 + 10 \times 2 + 10 \times 2) = 1160$ trials.

Results: Menu Application Study

Out of the 440 trials which allowed visual aid upon request, it was asked for in only 5 trials, showing that the visual to mapping was quickly understood by the users. This is of course aided by the alphabetical design of the menu.

Item Execution

The mean item execution accuracy % for 4 & 8 items over the three sessions, with visual aid, visual aid upon request, and without visual aid, is shown in Figure 7(a). For 4 items, accuracy remains relatively consistent at >95% for all three sessions. For 8 items, the accuracy is relatively lower, starting at 86.3% in the first session and ending at 92.5% in the third.

The mean item execution time is shown in Figure 7(c). For 4 items, execution time started out at 5.2s in the first session and improved to 3.1s in the final session. For 8 items, it went from 5.8s to 4.5s. The effect of session on execution time was significant for both. For 4 items, after Greenhouse-Geisser correction $F(1.129, 7.903) = 13.074, p < .01$. For 8 items, $F(2, 14) = 5.320, p < .05$. With <5 minutes of visual aid, not only were participants able to perform tactile direct manipulation in a tactile menu without visual aid, they improved upon their speeds while preserving an accuracy of >90%.

Drag & Drop

The mean drag & drop accuracy % for 4 & 8 items over the two sessions, visual aid upon request, and no visual aid, is shown in Figure 7b. The accuracy hovers in the range 85-90% for both 4 and 8 items. The mean item execution time

is shown in Figure 7(d). For 4 items, it goes from 11.6s to 9.9s. For 8 items, it goes from 13.3s to 12.3s. Even for drag & drop, these time values are high. We take a deeper dive into the participants' approach.

Results Discussion

We queried the participants on which of the three approaches they used for finding items: i) only *localization* - when they found items based only upon its location on the wrist, ii) only *counting* - when they only counted items to get to the desired one, or iii) using *Both* i) and ii). The terms were clearly explained to the participants. Figure 7(e) shows the distribution of their approaches for 4, 8 items. For 4 items, all 3 were used equally. For 8 items, no participant used *localization* alone, and more than half of them used *Both*. One participant remarked: "I knew that Dan was the bottom-mid item and felt it very clearly, so I reached Dan quickly and then counted from there." Multiple participants echoed similar strategies that involved using location and counting in tandem. In fact, all three participants who used *localization* for 4 items switched to using *Both* for 8 items, indicating that they found location to be an insufficient indicator when it came to a higher number of items. This shows that the use of *exploratory localization* in DMTs can overcome limitations posed by *localization*. However, our results are for an alphabetical menu which eases counting. The same performance and strategies might not apply without a logical ordering and needs to be studied.

We queried participants on each stage's mental and physical demand. Figure 7(f) shows the results. Unsurprisingly, execution with 4 items was the least mentally demanding, and drag & drop with 8 items, the most. Four items were easier than eight items for both tasks. However, even for 8 items, target execution's mental demand was low-to-medium. Given that both speed and accuracy of execution are also reasonable, target execution in such applications shows promise. Further, performance improves with practice. One participant remarked: "The concept was difficult to grab initially, but once I understood, I was surprised by the ease." Some participant remarks hinted at motor learning: "I started with counting the names but after some time I was not really counting, I just repeated what I did earlier without thinking."

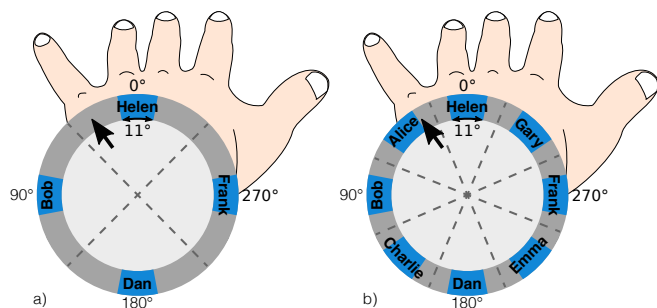


Figure 8: Study 3 Menu Layout: 4, 8 items. Blue regions are items. Dot partitions are activation zones for contained items. Pointer starts at 22°.

DISCUSSION

Contexts of Use

DMTs open up a new space of tactile interactions that enable adaptation of visual interface analogues for the tactile sense.

Rich actuation technologies in future wearables can be easily imagined. We already see instances such as smartwatches with traditional dials that only focus on tactile feedback [2]. We now list some routine scenarios of DMT use.

Private Interactions: Visual passcode entry procedures are vulnerable to over-the-shoulder glances. With a DMT, users can enter tactile passcodes that are secure in such situations.

Public Subtle Interactions: DMTs can aid public performers who need to interact with devices as part of their performance but prefer to keep visual attention towards their audience.

On-the-go Interactions: Users can perform eyes-free interactions while walking, running or driving. For example, when driving, the steering wheel can have an input controller for radio channels which are tactically located on the user's wrist. This requires an investigation of the limits of complexity the tactile interface can reach before users start getting significantly distracted from their primary activity. *Stealth Interactions:* In situations where users want to completely hide their device use, like a meeting or lecture, DMTs can be used stealthily.

The above scenarios are just to give a sense of DMT use. We're only scratching the surface of what DMTs are capable of, and the application scenarios can potentially range from routine wearable contexts to medical & visually impaired contexts, to unpredictable usecases like underwater apps.

Design Guidelines for DMTs

Based on our observations, we summarize nine design guidelines for building and interacting with DMTs. 1) Phantom sensations & frequency discrimination work well together to simulate tactile pixels and targets. 2) Only using location as a target discriminator works well provided exploratory localization is easy in the DMT. A precision counting study will be useful in this regard. 3) Exploratory localization is aided by targets positioned at anchor locations. 4) In the absence of strong anchors on the skin, the pointer start position is the default anchor and should therefore be reset to its start position every time the screen is switched on or a new application is opened, or even when a single interaction sequence is complete. This requirement might be relaxed in a non-circular display where the ends can act as anchors. 5) Maintaining a constant ratio of target:void widths helps the user discern targets and voids in regions of the skin where perception acuity is not as strong. 6) A logical ordering of targets on the screen eases exploratory localization and should be implemented where possible. 7) Even in non-menu applications, a buffer space around targets is useful to prevent marginally incorrect actions. 8) Since the Idle state has no response, users lose track of the interface state when clutching. The input needs to have minimal or no clutching to prevent loss of interaction continuity. 9) It will be useful to derive the performance model indices for various input devices for a particular DMT. This will help in selecting the apt input device.

An issue with extended duration tactile stimulations is that humans start feeling numb to it. However, in our system, the tactile response is not a single continuous tactile stimulation at a single frequency. The constant punctuation of low frequency voids with high frequency targets ensured that users did not get numb to a single tactile stimulation.

Limitations and Future Directions

The menu-study shows that the initial friction in use is overcome quickly. However, memory & attention demands of real world DMT tasks needs investigation. Our target acquisition study intended to establish the baseline dependencies of a no-distractor DMT task. It confirmed our assumptions about adherence to Fitts' width & distance and gave a sense of the performance of the wrist DMT. But, the exact performance model will not apply for a with-distractor design. Even, with distractors, the performance model will apply only to that particular layout of distractors on screen. It will still be useful to study a with-distractor design to see how the performance model compares. The no-distractor performance model, however, is a useful baseline to compare multiple DMTs, as well as multiple inputs for the same DMT.

Many extensions of current work are potential research directions. Interface paradigms other than indirect pointing and 2D DMTs are rich conceptual extensions. Other actuation technologies and skin regions can further shed light on DMT capabilities. Finally, advanced inputs such as single handed use for wrist DMTs can improve usability and performance.

CONCLUSION

We introduce the concept of direct manipulation for tactile displays and define DMTs - Direct Manipulation-enabled Tactile displays. To this end, we deconstruct the elements of a display and adapt them for tactile displays: tactile screen, tactile pixels, tactile pointer, and tactile targets. We describe the tactile indirect pointing interface that enables the user to track system state and perform user actions: tactile pointing, selection, execution, and drag & drop. We implement a proof of concept DMT that uses a unique combination of phantom sensations and frequency discrimination to generate a controllable pointer over a seemingly continuous tactile screen around the wrist using just 4 actuators and simulate targets different from voids. We define exploratory localization for DMTs and study and report its precision limits for our DMT which reasonably high at a target count of 19. We investigate the performance dependencies of target acquisition in DMTs and validate its adherence to Fitts' law. We found position to be an additionally significant factor. Based on the dependencies, we derive a performance model for our DMT. To demonstrate a DMT app analogous to visual apps, we study a 4, 8-item menu for target execution and drag & drop performance. With <5 minutes of visual aid, not only were participants able to perform tactile direct manipulation in a tactile menu without visual aid, they improved upon their speeds while preserving an accuracy of >90% for target execution. Drag & Drop was relatively harder to perform. The study evidenced the use of exploratory localization, more so for higher number of targets. We end with guidelines for DMT design.

Even with fundamental differences in our visual and tactile sense, the gap in visual and tactile interactions is too wide. Tactile direct manipulation is a pivotal step in reducing it.

ACKNOWLEDGMENTS

This project was supported by the European project Happiness, H2020 645145 and NSERC Canada, #482670.

REFERENCES

1. Stephen A Brewster and Alison King. 2005. The design and evaluation of a vibrotactile progress bar. In *World Haptics 2005*. IEEE, 499–500.
2. Matt Brian. 2015. Sony is crowdfunding a smart watch with a dumb face. (2015). <http://www.engadget.com/2015/08/31/sony-wena-smartwatch/>
3. William Buxton. 1990. A three-state model of graphical input. In *Proc. INTERACT'90*. 449–456. <http://billbuxton.com/3stateModel.pdf>
4. Marta G. Carcedo, Soon Hau Chua, Simon T. Perrault, Paweł Woźniak, Raj Joshi, Mohammad Obaid, Morten Fjeld, and Shengdong Zhao. 2016. HapticColor: Interpolating Color Information as Haptic Feedback to Assist the Color Blind. In *Proc. CHI '16*. 1–12. <http://dx.doi.org/10.1145/2858036.2858220>
5. Géry Casiez, Daniel Vogel, Ravin Balakrishnan, and Andy Cockburn. 2008. The impact of control-display gain on user performance in pointing tasks. *Human-Computer Interaction* 23, 3 (2008), 215–250. <http://dx.doi.org/10.1080/07370020802278163>
6. Hsiang-yu Chen, Joseph Santos, Matthew Graves, Kwangtaek Kim, and Hong Z. Tan. 2008. Tactor Localization at the Wrist. In *Proc. Eurohaptics'08*. 209–218. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.151.7152&rank=1>
7. Roger W. Cholewiak and Amy A. Collins. 2003. Vibrotactile localization on the arm: Effects of place, space, and age. *Percept. Psychophys.* 65, 7 (2003), 1058–1077. <http://www.springerlink.com/index/10.3758/BF03194834>
8. Sarah A Douglas, Arthur E Kirkpatrick, and I Scott MacKenzie. 1999. Testing pointing device performance and user assessment with the ISO 9241, Part 9 standard. In *Proc. CHI '99*. 215–222. <http://dx.doi.org/10.1145/302979.303042>
9. James J. Gibson. 1962. Observations on Active Touch. *Psychological Review* 69, 6 (1962), 477–491. <http://psycnet.apa.org/psycinfo/1963-06036-001>
10. Ali Israr and Ivan Poupyrev. 2011. Tactile Brush: Drawing on Skin with a Tactile Grid Display. In *Proc. CHI'11*. 2019–2028. <http://doi.acm.org/10.1145/1978942.1979235>
11. Yeon Sub Jin, Han Yong Chun, Eun Tai Kim, and Sungchul Kang. 2014. VT-ware: A wearable tactile device for upper extremity motion guidance. In *Proc. RO-MAN'14*. 335–340. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6926275>
12. Seungyon "Claire" Lee and Thad Starner. 2010. BuzzWear: alert perception in wearable tactile displays on the wrist. In *Proc. CHI'10*. 433–442. <http://dl.acm.org/citation.cfm?id=1753326.1753392>
13. Charles Lenay, S. Canu, and P. Villon. 1997. Technology and perception: the contribution of sensory substitution systems. In *Proc. ICCT'97*. 44–53. <http://dx.doi.org/10.1109/CT.1997.617681>
14. Shachar Maidenbaum, Sami Abboud, and Amir Amedi. 2014. Sensory substitution: closing the gap between basic research and widespread practical visual rehabilitation. *Neuroscience & Biobehavioral Reviews* 41 (2014), 3–15.
15. Michael Matscheko, Alois Ferscha, Andreas Riener, and Manuel Lehner. 2010a. Tactor placement in wrist worn wearables. In *Proc. ISWC'10*. 1–8. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5665867>
16. Michael Matscheko, Alois Ferscha, Andreas Riener, and Manuel Lehner. 2010b. Tactor placement in wrist worn wearables. In *Int. Symp. Wearable Comput. 2010*. IEEE, 1–8. DOI : <http://dx.doi.org/10.1109/ISWC.2010.5665867>
17. Hugo Nicolau, João Guerreiro, Tiago Guerreiro, and Luís Carriço. 2013. UbiBraille: designing and evaluating a vibrotactile Braille-reading device. In *Proc. ASSETS'13*. 1–8. <http://dx.doi.org/10.1145/2513383.2513437>
18. Jerome Pasquero, Scott J. Stobbe, and Noel Stonehouse. 2011. A haptic wristwatch for eyes-free interactions. In *Proc. CHI'11*. 3257–3266. <http://dl.acm.org/citation.cfm?id=1978942.1979425>
19. L. Rahal, Jongeun Cha, A.E. Saddik, J. Kammerl, and E. Steinbach. 2009. Investigating the influence of temporal intensity changes on apparent movement phenomenon. In *Virtual Environments, Human-Computer Interfaces and Measurements Systems, 2009. VECIMS '09. IEEE International Conference on*. 310–313. DOI : <http://dx.doi.org/10.1109/VECIMS.2009.5068914>
20. Christian Schönauer, Kenichiro Fukushi, Alex Olwal, Hannes Kaufmann, and Ramesh Kaskar. 2012. Multimodal motion guidance: techniques for adaptive and dynamic feedback. In *Proc. ICMI'12*. 133–140. <http://dl.acm.org/citation.cfm?id=2388676.2388706>
21. Ben Shneiderman. 1983. Direct Manipulation: A Step Beyond Programming Languages. *Computer* 16, 8 (1983), 57–69. <http://dx.doi.org/10.1109/MC.1983.1654471>
22. Georg von Békésy. 1957. Sensations on the Skin Similar to Directional Hearing, Beats, and Harmonics of the Ear. *J. Acoust. Soc. Am.* 29, 4 (1957), 489–501. <http://psycnet.apa.org/doi/10.1121/1.1908938>
23. Cheng Xu, Ali Israr, Ivan Poupyrev, Olivier Bau, and Chris Harrison. 2011. Tactile display for the visually impaired using TeslaTouch. In *Ext. Abs. CHI'11*. 317–322. <http://dx.doi.org/10.1145/1979742.1979705>