Holes, Valleys, and Pits: Using Data-Morphed Topographies to Improve Precision of Touchless Input

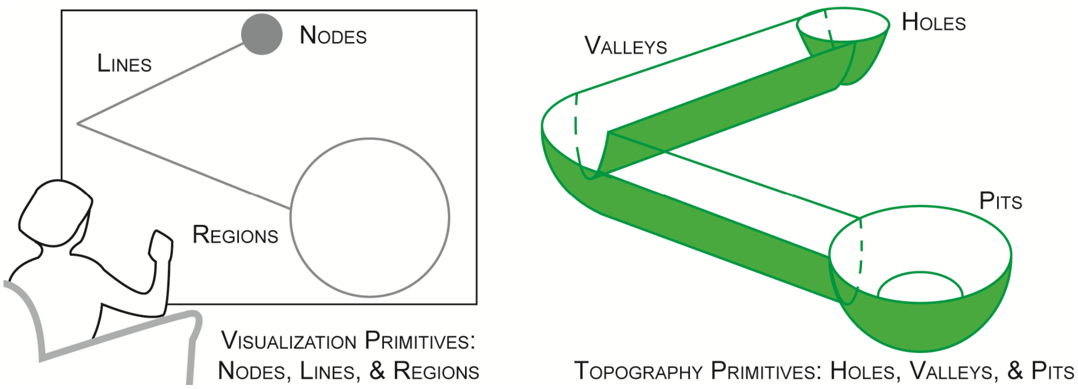

Figure 1. Data-Morphed Topographies (DMTs) are virtual surfaces (e.g., holes, valleys, or pits) that overlay on visualization structures (e.g., nodes, lines, or regions) and modify cursor movements to improve the precision of touchless input during data browsing.

ABSTRACT

Large, high-resolution displays enable efficient visualization of large datasets. To interact with such visualizations, touchless interfaces can potentially support fluid interaction at different distances from the display. But touchless gestures lack haptic feedback, thereby causing users' gestures to unintentionally move off interface elements and requiring additional effort to perform accurate actions. To mitigate this problem, we introduce Data-Morphed Topographies (DMTs): adaptive constraints on cursor movements that guide touchless interaction along the structure of visualized data. We present three topography primitives—*holes*, *valleys, and pits*—which map to common visualization primitives, such as nodes, lines, and regions. In a controlled comparison with unconstrained touchless input, participants were significantly more precise with DMTs, and did not perceive significantly more workload. The efficacy of DMTs was significantly greater in more difficult tasks. By morphing data visualizations into virtual topographies, our work is a crucial first step to improve the precision of touchless input.

Author Keywords

Touchless Input; large display; mid-air gestures; pseudo-haptic textures; natural user interfaces.

ACM Classification Keywords

H.5.2 [User Interfaces]: Interaction styles.

INTRODUCTION

Epidemiologists Joan and Zeb are exploring a wall-sized digital map that details the spread over time of the recent flu outbreak across five different counties. As Joan sits back a few feet from the display, she gradually gestures within a circle that marked a less affected area, and quickly visualizes the types of messages received at the emergency operation center on the sidebar. To find any possible pattern, as she examines the messages, she fails to realize that her hand has been moving off to adjacent regions, thus displaying unrelated messages.

In an increasingly data-oriented world, collaboration and decision making continues to thrive around effective data visualization [7]. To better acclimatize this, data visualization researchers are exploring novel interaction displays, modalities, and techniques—beyond the desktop [6]. Compared with desktops, large, high-resolution displays support efficient visualization of large, scientific datasets, effective sensemaking, difficult data manipulations [14], and seamless co-located collaboration [11]. The ample physical navigation afforded by large displays calls for post-WIMP interaction modalities, such as multitouch [11], mid-air devices [15], or touchless [7].

Touchless interfaces enable users to interact with large displays from a distance, thus delivering a panoramic view as they stand away or sit back during their interactions. By affording physical navigation and interaction proxemics, touchless gestures show potential to support a fluid engagement with interactive information visualizations (InfoVis) [7]. For example, panning, zooming, or exploring details on demand are typical InfoVis interactions [21] that lend themselves well to touchless gestures. However, device-less touchless input lacks any haptic feedback and requires users to depend solely on their proprioception (one's sense of position and orientation of body parts) [15]. Thus, touchless gestures add abundant fluidity to an interaction technique, but fail to provide fine-grained, pixel-level, motor guidance for accurate target acquisitions.

To improve the precision of touchless input, specifically for interacting with data visualizations on large displays, our paper makes the following main contributions:

- We introduce Data-Morphed Topographies (DMTs): spatial configurations (e.g., holes, valleys, and pits) virtually overlaid on visualization structures (e.g., nodes, lines, and regions) that constrain the touchless cursor's imprecise movements during data browsing—conveniently along the structure of the visualization.
- We implemented the three topography primitives, holes, valleys, and pits, and introduced two techniques to augment their effectiveness, *adaptive* and *additive* topographies. Adaptive topographies dynamically adapt their shapes to constrain imprecise movements, and additive topographies combine multiple primitives to suit a specific data-visualization structure.
- Finally, we empirically evaluated DMTs in a controlled study with 17 participants performing tasks of varying difficulty. DMTs significantly improved the precision of touchless input compared with unconstrained touchless without significantly increasing perceived workload. Our results also indicated that the efficacy of DMTs was significantly greater in more difficult tasks.

By morphing a data visualization into a dynamic, virtual topography, our work is a crucial first step to combat the imprecision of touchless input during InfoVis interactions.

RELATED WORK

As data visualization tasks continue to grow in pixels (big datasets) and people (collaborative interaction), researchers have set out to explore the interplay of interaction and interaction displays in *beyond-the-desktop* settings [5]. For example, large, high-resolution, vertical displays are being studied for collaboration [11], data manipulation [14], or proxemic interactions around data visualizations [7]. This current research on large display interaction distinguishes itself from the past decade [17] in chiefly exploring post-WIMP interaction modalities, such as multitouch [11, 14], mid-air, or touchless [7]. However, such large-display interactions with visualizations have mostly focused on panning and zooming interactions [7, 15], i.e., *explore* and *abstract/elaborate* type of InfoVis interactions [21]. Equally relevant, but less explored is another abstract/elaborate interaction technique—*details-on-demand*. For instance, in large-display touchless interactions, how would we emulate the simple tool-tip interaction that provides detailed information when a mouse cursor hovers over a data item? More crucially, how does this shift in input device—from mouse to bare hands—may affect users' interaction experience?

Markerless depth sensors and computer vision techniques have made it feasible to interact with distant displays using touchless gestures [7]. However, due to the absence of haptic feedback, touchless interactions have been found less efficient and more fatiguing than device-based gestures [15]. Touchless performance also depends on the type of interaction primitives involved. For example, motor-intuitive touchless gestures allow users to apply their pre-existing sensorimotor knowledge unconsciously [5]. User interface (UI) controls employing motor-intuitive primitives, such as touchless menus, were found more efficient than UI controls based on system-defined primitives [2]. However, imprecision in touchless input and thus an increased effort to perform accurate touchless manipulations continue to impede the progress of touchless interactions with InfoVis.

To mitigate this problem, prior solutions proposed using device-based, mid-air input for large display interactions [15]; with these mid-air devices tailor-made and *connected* to the interface (e.g., Wii, Gyro mouse, or tablet). Other works explored the use of any available device as a *mobile token*—not digitally connected, but expected to provide motor guidance by virtue of its tangibility [1]. All these approaches outlined above propose improving touchless interactions by orchestrating users' mid-air input. We, on the other hand, aim to improve the precision of touchless input by shifting the burden from users' input to the user interface. Prior work similar to us proposed air vortex as haptic feedback for touchless interactions, but did not evaluate their effect on user performance [18]. Our approach, however, is less physical and more virtual—designing the user interface software to improve touchless precision. To that aim, we build upon three research areas: pseudo-haptic textures, kinematic templates, and mouse cursor guides for data browsing.

Pseudo-Haptic Textures

Inspired by haptic interfaces, researchers have proposed modifying the shape [13] or the motion of the mouse cursor to simulate pseudo-haptic textures [12]. Pseudo-haptic textures that vary the cursor's motion (using a transfer function) aim to generate the effect of lateral forces when passing over a texture. Such lateral forces were found to dominate other perceptual cues when feeling textures with 2D force-feedback devices [16]. Similarly, with a mouse, users were able to perceive pseudo-haptic textures, such as *bumps* and *holes*—with or without visual feedback [12]. We not only introduce pseudo-haptic textures in touchless, but also propose semantically morphing data visualizations to such textures and explain how such data-morphed topographies can improve touchless input precision.

Kinematic Templates

Amplifying or dampening the mouse cursor's speed has been used previously in other applications, such as *kinematic templates* for content-relative cursor manipulations [8]. Kinematic templates are user-defined regions that reinterpret pointer input (e.g., a stroke) to guide digital drawing (e.g., drawing parallel lines or circles). Instead of *user-defined* regions for point-and-click cursor manipulations, our approach *morphs* a data-visualization into a virtual topography, thereby semantically modifying the touchless cursor's speed during data browsing.

Cursor Guides for Data Browsing

To simulate pseudo-haptic sensations, instead of modifying the transfer function, prior work proposed directly adding tiny displacements to the mouse cursor [20]. Such *active cursor displacements* were then illustrated in a decision graph animation that would push the cursor along the structure of the graph (i.e., its nodes and directed edges)[1]. For semantically unstructured browsing, such as scrolling, *content-aware scrolling* introduced dynamic paths and variable cursor speeds to represent the relative importance of different regions within a document [10].

Learning from such crucial prior work, in what follows, we present our approach to increase the precision of touchless input while browsing data visualizations on large displays.

DATA-MORPHED TOPOGRAPHIES (DMT)

Data-Morphed Topographies (DMTs) are special regions within a data visualization that manipulates touchless input (or touchless cursor) to improve interaction precision [4]. Touchless cursor manipulation is achieved by adjusting the Control/Display[2] (C/D) ratio with a transfer function. This transfer function is formulated according to the geometrical structure of the visualization, i.e., the data visualization is morphed into a virtual topography. By morphing the visualization with a virtual terrain that is semantic with the visualization's structure (e.g., rows, columns, or regions), DMTs constrain users' imprecise touchless movement during data browsing.

To implement DMTs, we build upon prior work on *height maps*. Height maps can vary the mouse cursor's speed to conjure up a feeling of travelling over uneven topographical surfaces [12]. To simulate different topographies, we first propose topography primitives and discuss their design parameters. We then introduce two techniques to augment the effectiveness of DMTs: *adaptive* and *additive* topographies. Finally, we describe a simple visual feedback routine that abstractly conveys the model used in our approach to constrain users' touchless input.

[1] http://www.powercursor.com/

[2] Control/Display ratio (C/D ratio) is the ratio of the speed of the hand movement (control) to the speed of the cursor movement (display).

```
AmPx ← Amount of pixels moved in control space
PrevPos ← Previous position in display space
CurrPos ← Current position in display space
T ← Topography constant

ApplyTopography (PrevPos, CurrPos, AmPx)
DO
   NextPixel ← CalcNextPx (PrevPos, CurrPos)
   DiffHeight ← CalcDh (PrevPos, NextPixel)

   IF DiffHeight > 0
          CostOfMovement ← 1 + T × |DiffHeight|
   ELSE
          CostOfMovement ← 1 – T × |DiffHeight|
   ENDIF
   IF AmPx > CostOfMovement
          PrevPos ← NextPixel
          AmPx  ← AmPx – CostOfMovement
   ELSE
          CurrPos = PrevPos
   ENDIF
WHILE AmPx > CostOfMovement
RETURN CurrPos

CalcNextPx (PrevPos, CurrPos)
   RETURN the next pixel given the previous pixel and
   a theoretical path (current pixel) using the 8
   nearest-neighbor algorithm

CalcDh (PrevPos, NextPixel)
   RETURN height difference between two consecutive
   pixels based on the Height Map
```

Figure 2. Algorithm for travelling height maps (based on [12]).

Height Maps: Simulating a Topography

The topography of a surface is a function of different heights—a *height map*. So a topography can be simulated on a user interface by maintaining a height value associated with each pixel of the screen. A slope can be simulated using either a Gaussian profile or a Polynomial profile (e.g., **Figure 4**). We used the algorithm proposed in [12] to implement a topography (**Figure 2**): Users' movement in the control space is mapped to the touchless cursor's movement in the display space as a function of the height map of the topography. The *cost of displacement* between two consecutive pixels is determined by their difference in height. When this difference in height is negative, the cost is greater than 1 (i.e., user has to move more in the control space than usual, or *ascend*) and when the height difference is positive, the cost is less than 1 (i.e., user moves less in the control space than usual, or *descends*). Until users' movement in control space exceeds the cost of displacement, the touchless cursor is constrained at its prior position, thus simulating a virtual terrain.

Topography Primitives: Holes, Valleys, and Pits

To match the geometry of common visualization primitives, nodes, lines, and regions, we propose using *height maps* (**Figure 3**) to simulate *Holes*, *Valleys,* and *Pits* (**Figure 1**).

Holes: A hole is a narrow, circular depression from a baseline plane that is simulated using mathematical profiles, such as a Gaussian, a polynomial, or a linear profile [12]. For example, a vertical cross-section of a Gaussian Hole can be computed as

$$H = H_{max} \times \exp(-x)^2,$$

where H = height of the pixel x, and H_{max} = height of the baseline plane.

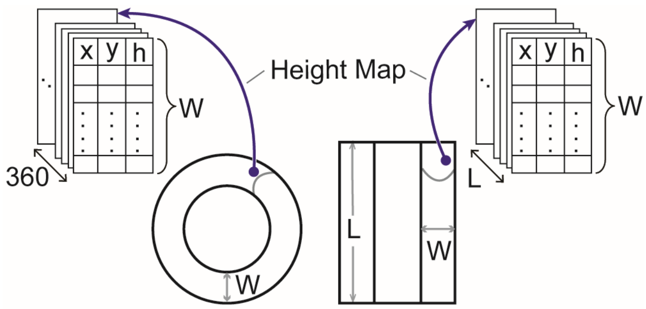

Figure 3. A vertical cross-section of a pit or a valley is stored as a height map, with h = $H_{max} \times f(step)$, ∀ step: step ∈ W.

Valleys: A valley is a linear depression from a baseline plane. Similar to holes, valleys are simulated with a Gaussian or a polynomial profile. The vertical cross section of a valley is either similar to a Hole (a V-shaped valley) or to a Pit (a U-shaped valley). Such variations are different subtypes of a topography primitive, as they correspond to a single visualization primitive (**Figure 4**).

Pits: A pit is a wide, circular depression from a baseline plane whose left slope is simulated using an exponential decay function, $H = H_{max} \times \exp(-x)$, and right using an exponential growth function, $H = H_{max} \times \exp(x)$.

To simulate valleys and pits, we chose a polynomial profile:

```
FOR each pixel P along the length of the wall
    H = Hmax × exp(− Slope × P)
ENDFOR
```

Topography Parameters

Overall, our three topography primitives constitute of the following four parameters: *Wall Length*, *Slope*, H_{max}, and T. A higher value of T amplifies the slope of a topography and increases or decreases the cost of displacement (see **Figure 2**). Through iterative tuning, we identified an optimum range of T as [100, 500]. A combination of the parameters *Wall Length* and H_{max} play the same role as the parameter *Slope* in simulating a steep or gradual ascent/descent.

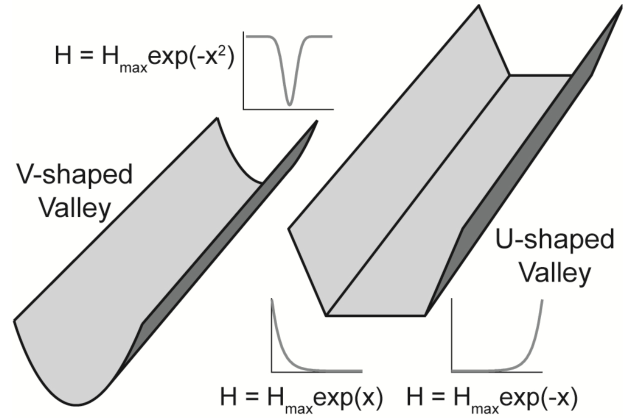

Figure 4. Two different types of valleys: V-shaped and U-shaped. (H = current height, H_{max} = maximum height)

For our final experiments with DMTs, we used H_{max} = 10, *Wall Length* = 5, *Slope* = 0.1, and T = 400 (valley) /200 (pit). In our parameter tuning phase, we encountered the following phenomenon: When users move obliquely to the wall of any topography, they took a long-winded path to get out; but was less constrained than when moving orthogonally to the wall. To mitigate this problem, we introduced a new technique to effectively constrain touchless input—*Adaptive Topography*.

Adaptive Topographies

Slope of a topography allows a gradual descent into a hole, a pit, or a valley. However, during getting out, the ascent—that ultimately constraints the user input—depends on how users move along the wall of the topography: an orthogonal movement provides the designed resistance, but an oblique movement lacks constraint due to small differences in height while traversing along the wall (similar to taking the ramp instead of climbing a huge step). Thus, we introduce *adaptive topographies* (**Figure 5**). After users successfully enter a pit or a valley, the inclined walls adapt to a vertical shape, thereby preventing any oblique user movements to

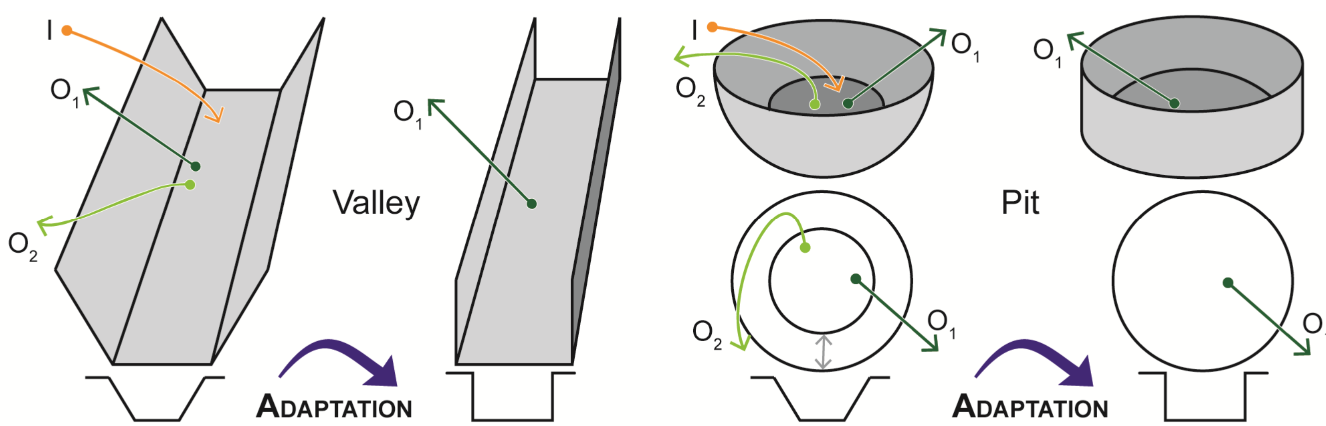

Figure 5. Slope of a valley (or a pit) allows users to gradually move into the topography (I). To get out, users can move orthogonally (O1) or obliquely (O2) relative to the wall of the valley (or the pit). Due to small differences in height, however, a long oblique movement along the wall (O2) fails to sufficiently constrain users' touchless input. To mitigate this, we introduce *Adaptive Topography*: after users enter a valley (or a pit), its walls shape-shift to become vertical, thus requiring a higher cost of displacement to move out of the topography, and thereby appropriately constraining users' touchless input to the region.

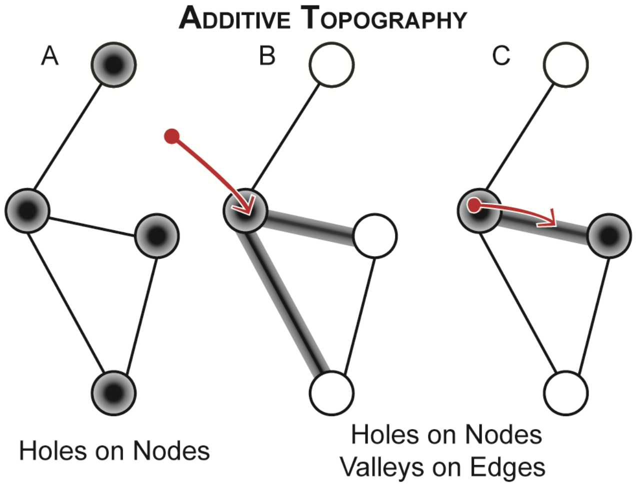

Figure 6. ***Additive Topography*** **combines multiple topography primitives. For example, holes and valleys can be overlaid on a graph and dynamically invoked to constrain touchless input. At first, a graph has holes overlaid on all its nodes (A); as a node is reached, valleys are invoked on its edges and holes dismissed from other nodes (B); as an edge is traversed, its endpoints are overlaid with holes (C).**

unwittingly leave the topography during data browsing. Holes, however, do not require any adaptation, because they map to nodes that do not require detailed interaction within the topography—instead, holes play the role of transition stops between interconnected paths.

Additive Topographies

Similar to interaction primitives constituting interaction controls, topography primitives can be combined together to match non-trivial data visualizations. To that aim, we introduce *Additive Topographies*. Because the complex nature of additive topographies may over constrain users' touchless interaction, we suggest dynamic invocation of primitives in such scenarios, thus fostering a seamless data browsing experience. For example, a graph (with nodes and edges) can be morphed into a set of holes and valleys (**Figure 6**). At first the graph would only contain holes to allow a flexible starting point for users. As users reach a node, valleys would be invoked on its connecting edges; and as a particular edge (with valley) is being traversed, only its endpoints would contain holes.

Visual Feedback

DMTs are *invisible* user interface (UI) techniques. Prior research found that users are able to perceive these primitives without any visual feedback [12]. The objective of DMTs is not to primarily cause a feeling of bumps and holes, but to use such perceptions to constrain users' touchless input. Based on our algorithm (**Figure 2**), the touchless cursor—while ascending out of a topography—ceases to move until sufficient displacement occurs in the control space. When this happens, users invariably make longer and faster movements to unchain themselves. While this process is principal toward the perception of a topography, it fails our purpose of constraining user input. To mitigate this problem, we introduced a visual feedback routine that loops in users to actively use DMTs and constrain imprecise movements. When users' touchless cursor begins *ascending* out of a DMT, two visual cues appear: an additional cursor representing users' position in the control space, and a bright *trail* connecting the two cursors. Width of this trail represents the current *cost of displacement* to exit the topography (see **Figure 2**). As users successfully exit a DMT, the trail snaps and the second cursor disappears. Our feedback routine plays two roles. First, users can employ this visual cue to actively get back in their region of interest. Second, by gauging the trail's width, users can estimate the additional movement required to exit a DMT.

EVALUATION: USER PERFORMANCE & PREFERENCE

We designed Data-Morphed Topographies (DMTs) to improve the precision of touchless input. To test the efficacy of DMTs in large-display, touchless interactions, we evaluated the user experience of a common InfoVis interaction: *details on demand* (or abstract/elaborate [21]).

We investigated two research questions (RQs). First, in a controlled study with abstract tasks, we wanted to assess how a DMT affects users' efficiency, accuracy, and perceived workload (*RQ1*). We compared DMTs with unconstrained touchless input and token. Prior work on *device-agnostic*, touchless input suggests an advantage of using any available device as a token in touchless interactions because of its inherent tangibility [1]. So we explored four types of GUIDANCE: *Free* (or no guidance), *Token*, *DMT*, and *DMT & Token*. Second, we wanted to understand users' preference for DMTs as constraints for touchless input (*RQ2*). Prior to reporting such preferences, users interacted with a realistic InfoVis task using touchless gestures—with and without DMTs—thereby improving the ecological validity of their responses.

Hypothesis

Prior work reports a lack of precision in unconstrained touchless interaction [2, 15]. To improve the accuracy of touchless input without penalizing efficiency or overall workload, we introduced DMTs. To examine *RQ1,* we hypothesized the following:

H1: DMT will not affect the efficiency of touchless input.

H2: *DMT will increase the accuracy of touchless input*.

H3: *The more precision is required, the more DMT will increase the accuracy of touchless input.*

H4: *DMT will not affect the workload of touchless input.*

Method

Participants

We recruited 17 paid participants from a Midwestern public university in USA (M_{age} = 24.31, SE_{age} = 1.51, 7 females). All participants were right-handed and 14 were familiar with motion-tracking sensors: Kinect, Wii, or Leap Motion.

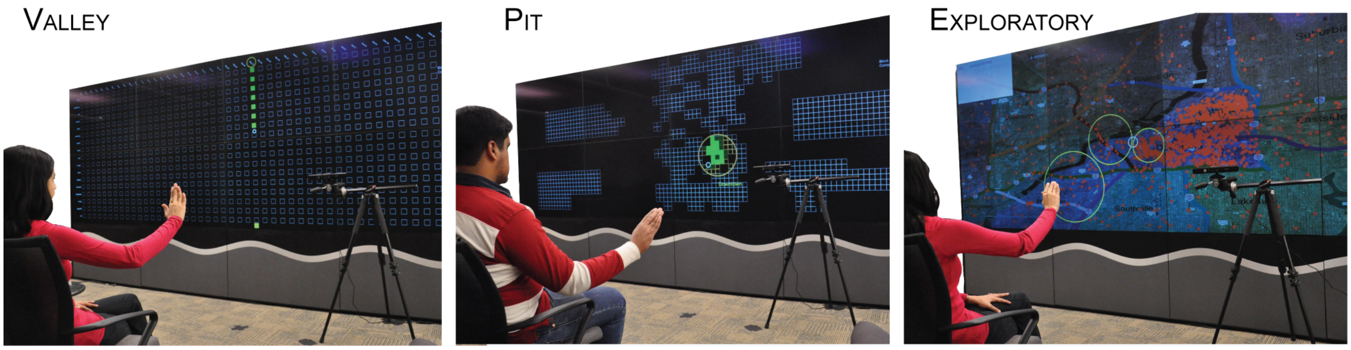

Figure 7. Participants performed details-on-demand interactions on a large display, including a contiguous grid of low density in the easy condition of Task 1 (left), a contiguous non-grid of high density in the difficult condition of Task 2 (center), and an exploratory prototype, based on the VAST 2011 dataset Epidemic Spread (right) [9].

Apparatus

We used a 160" wide by 60" high display with over 15.3 million pixels. The display, integrated by Fakespace Systems, comprises of eight 50" projection cubes (each with a resolution of 1600 x 1200 pixels) laid out in a 4 x 2 matrix, and is driven by a single computer. Instead of submillimeter-accurate sensors, we evaluated DMTs using off-the-shelf hardware—a Kinect™ for Windows. All experiments were written in C#/WPF running on Windows 7, and were implemented with Windows Kinect SDK 1.8.

Tasks and Procedure

To test our hypothesis, we designed two abstract tasks representing details-on-demand interaction [21]. Task 1 emulated interacting with a *heat map*—a rectangular grid of contiguously-spaced cells (**Figure 7**, left). Task 2 emulated interacting with *information bubbles*— contiguously-spaced cells in a non-grid (**Figure 7**, center). In task 1, participants traversed the target column—the region of interest (ROI)— from the topmost to the bottommost cell (64px-square). As a cell in the ROI (ROI cell) was passed, it turned green to symbolize success; when a non-ROI cell was traversed, the cell flashed red to indicate an error. In task 2, participants had to pass over each cell in the ROI (576 px diameter circle) at least twice, which first turned yellow and then green. Passing a cell twice represented the typical repetitive interaction during processing information from a data visualization. Overall, task 1 required more controlled user input than task 2 because of its implicit spatial complexity.

Furthermore, within each task (task 1 and task 2), we manipulated the input precision required, or task difficulty. Task difficulty was operationalized as spatial density: an *easy* task was half as densely populated as a *difficult* task. Overshoots occurred when participants moved out of the ROI. When all cells within the ROI turned green, a trial ended flashing a SUCCESS message (a trial continued until completed successfully). In a repeated-measures within-subject experiment, we measured task completion time (for efficiency) and number of overshoots (for accuracy) for each of 816 trials: 2 tasks × 4 GUIDANCE × 2 DIFFICULTY × 3 ROIs × 17 participants. Trials and tasks were completely randomized within subjects and across subjects.

Participants sat in a (21" high) chair, 2 m from the large display (~ 1.5 m from the sensor) and took about an hour to complete the study. Participants' movements were mapped from the control space to the display space as 1: 3.75 (the baseline C/D ratio). Trials were video recorded. Prior to each task, on an average all participants practiced three trials at each ROI with *DMT*. Participants rested at least 10s after every 3 trials and 10 minutes after completing all trials of a task. After completing each task (all trials), participants self-reported their perceived workload using the NASA-TLX instrument. To prevent over-exposing participants to the instrument (16 times), we only measured workload for *Free*, *Token*, and *DMT* across *easy* and *difficult* conditions (6 times). We also logged task completion times, number of overshoots from the ROI, and trajectory paths for each trial. Time, overshoots, and paths were measured from the first time participants landed on the ROI to until the completion of the trial. Task completion times included the time spent in overshooting from the ROI and subsequently recovering. Overshoots from the ROI that were more than 500 pixels away were discarded as system errors.

Exploratory Task. Abstract tasks serve well the purpose of a tight control in assessing users' objective performance, but they lack the *look and feel* of an interaction technique in a real-world context. So to acquire a better vision about the use of DMTs—prior to sharing their interaction preferences (*RQ2*)—participants explored a low-key, interactive visualization prototype: *Epidemic Spread* (see **Figure 7**, right). *Epidemic Spread* was designed using the dataset from the VAST 2011 challenge (a city map, text messages, and metadata [9]). Point of origin of posts with at least one keyword related to the epidemic or events leading to the epidemic were shown on the map. As participants browsed over the map, a word cloud displayed all the keywords shared from the positions underlying the participants' cursor (128 px square, in all interaction conditions); the font size of a keyword indicated its frequency. We defined *three* ROIs on the map. Using the word cloud, participants tried to figure out one major event that occurred in each of those ROIs. During this task (~ 20 minutes), they used *Mouse*, *Free*, *Token*, and *DMT* in no particular order.

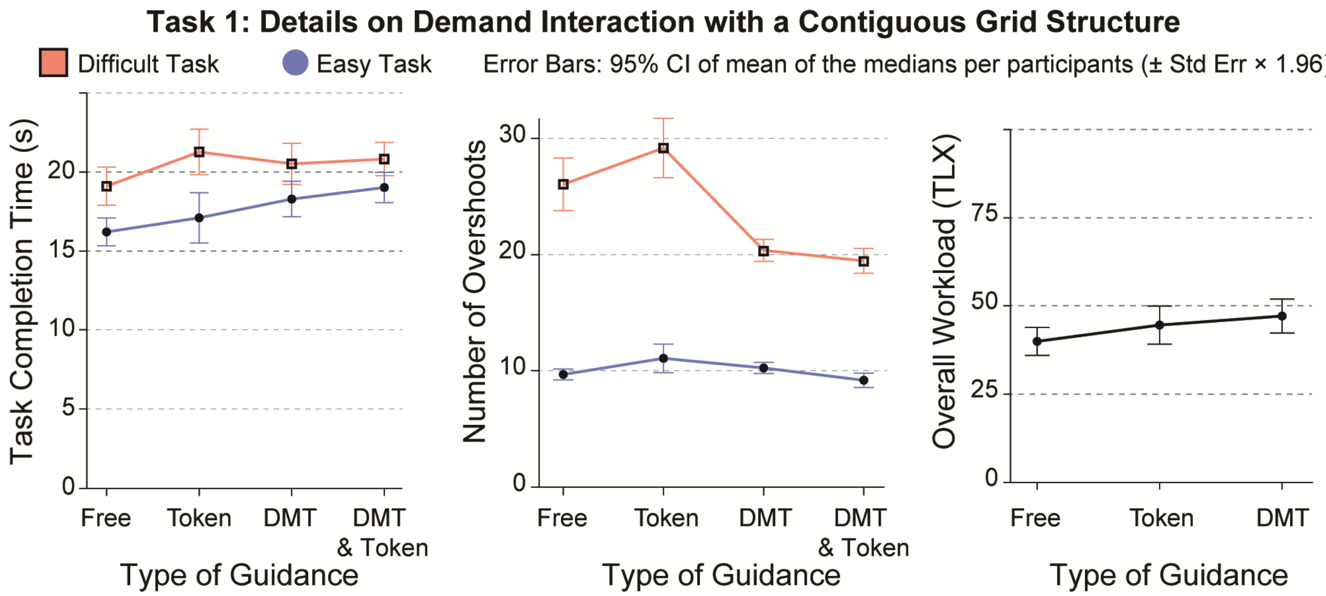

Figure 8. For Task 1, without any significant loss in efficiency or significant increase in perceived workload, Data-Morphed Topographies (DMTs) significantly reduced the number of overshoots, thus improving participant's precision of touchless input.

Finally, after engaging in the exploratory task, participants reported their perceived workload (for *Mouse* vs. *DMT*), interaction preferences, and shared overall comments.

RESULTS

For all 17 participants, task completion time and number of overshoots were perfectly correlated with the overall performance measures. As expected, task completion times were positively skewed; thus replications of unique experimental conditions were represented by their median. Our analysis used GLMM with standard repeated measures REML technique [2]. Participant was handled as a random factor. We report *F*-statistic using type III ANOVA with *Satterthwaite* approximation, and pairwise comparisons (using pooled variance) with *Holm-Bonferroni* correction. We found a learning effect across blocks: For the difficult task, participants performed about 3.7s more slowly in the first block of task 1 than the last block (2.8s more slowly for task 2). As the factors, GUIDANCE and DIFFICULTY, were counter-balanced, this did not adversely affect our analysis.

Task 1: Interaction with a Contiguous Grid

Efficiency

We found significant main effects of DIFFICULTY, $F(1, 112) = 23.27$, $p < .001$, and GUIDANCE, $F(3, 112) = 2.87$, $p = .039$, but no significant interaction effect (**Figure 8**, left). Participants took significantly more time to complete the difficult task ($M = 20$s, $SD = 5.13$) than the easy task ($M = 18$s, $SD = 4.84$), $p < .001$, $r = 0.50$, which confirmed our manipulations of task difficulty. Pairwise comparisons did not find any significant effect of *DMT* on efficiency (*Free* vs. *DMT* or *Token* vs. *DMT & Token*). *H1* was supported.

Accuracy

We found significant main effects of DIFFICULTY, $F(1, 112) = 199$, $p < .001$ ($M_{high} = 23.77$, $SD_{high} = 8.47$; $M_{low} = 10.04$, $SD_{low} = 3.15$), and GUIDANCE, $F(3, 112) = 7.20$, $p < .001$, and an interaction effect of GUIDANCE × DIFFICULTY, $F(3, 112) = 4.48$, $p = .005$ (**Figure 8**, center). Pairwise comparisons indicated that in the *difficult* task, participants made significantly fewer overshoots with *DMT* ($M = 20.38$, $SD = 3.90$) than *Free* ($M = 26.06$, $SD = 9.30$), $p = .031$, $r = 0.51$, and significantly fewer overshoots with *DMT & Token* ($M = 19.47$, $SD = 4.40$) than *Token* ($M = 29.18$, $SD = 10.53$), $p < .001$, $r = 0.72$. No significant results were found for the *easy* task. Post hoc Tukey-tests did not find significant differences between *Free* and *Token* for either *easy* or *difficult* task. *H2* was partially supported—only for the *difficult* task. *H3* was supported.

Satisfaction

We found no significant effect of GUIDANCE on *overall workload*, $p = .132$ (**Figure 8**, right). However, GUIDANCE significantly affected perceived *effort*, $p = .025$, but not perceived *performance*, $p = .793$. *H4* was supported.

Task 2: Interaction with a Contiguous Non-Grid

Efficiency

We found a main effect of DIFFICULTY, $F(1, 112) = 459$, $p < .001$, but no significant effect of either GUIDANCE or of the GUIDANCE × DIFFICULTY interaction (**Figure 9**, left). Participants took significantly more time for the *difficult* task ($M = 29$s, $SD = 6.55$) than the *easy* task ($M = 16$s, $SD = 2.82$), $p < .001$, $r = 0.92$. *H1* was supported.

Accuracy

Only DIFFICULTY had a significant effect on the number of overshoots, $F(1, 112) = 132$, $p < .001$ ($M_{high} = 30.43$, $SD_{high} = 17.62$; $M_{low} = 5.51$, $SD_{low} = 3.27$) (**Figure 9**, center). In the difficult task, participants made more overshoots in *Token* ($M = 33.12$, $SD = 14.64$) than *DMT & Token* ($M = 24.59$, SD=19.59), with results

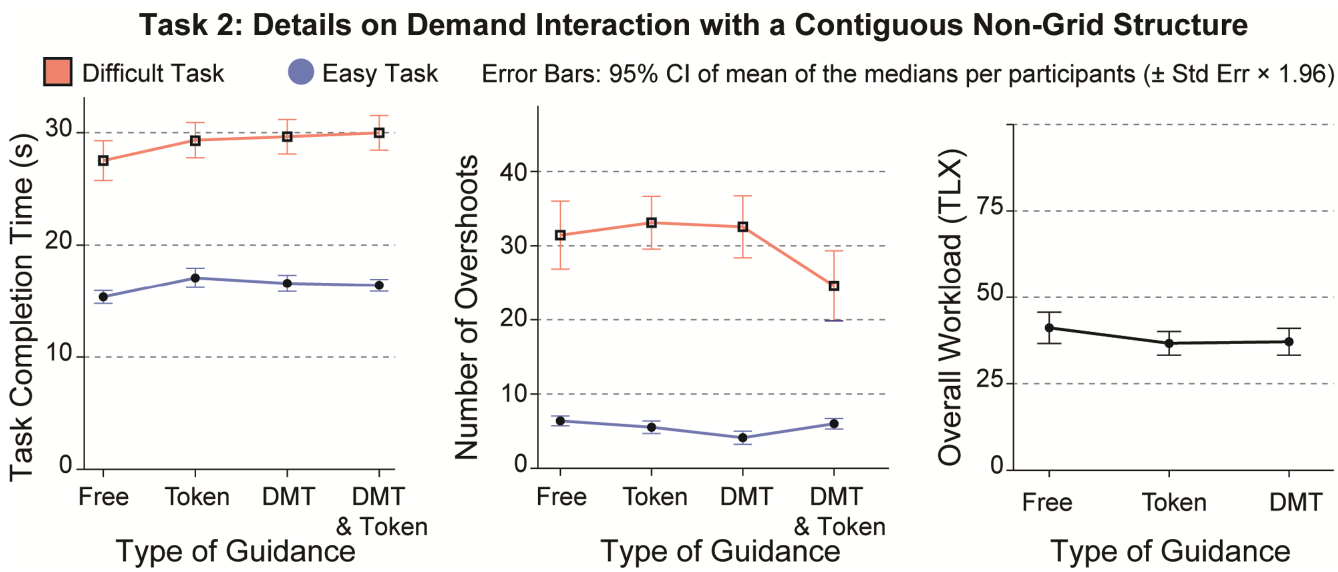

Figure 9. For Task 2, Data-Morphed Topographies (DMTs) did not significantly affect efficiency or perceived workload; but participants made fewer overshoots with DMT & Token than Token alone, with results approaching significance, p = .056.

approaching significance, $p = .056$. Similar to Task 1, post hoc tests did not find significant differences between *Free* and *Token* for either *easy* or *difficult* task. Neither *H2* nor *H3* was supported.

Satisfaction

We found no significant effect of GUIDANCE on participants' *overall workload*, $p = .292$ (**Figure 9**, right). Furthermore, GUIDANCE neither significantly affected participants' perceived *effort*, $p = .708$, nor perceived *performance*, $p = .902$. *H4* was supported.

Performance across Task 1 and Task 2

To explore the effects of GUIDANCE and DIFFICULTY on performance across tasks, we fitted a hierarchical mixed-effects model with participant as a random factor, task as a random factor, and task difficulty nested within the task factor: TIME/OVERSHOOT ~ GUIDANCE × DIFFICULTY + rand(PARTICIPANT) + rand(TASK/DIFFICULTY).

Our fitted model found a main effect of GUIDANCE on task completion time, $F(3, 246) = 3.80$, $p = .010$, but no interaction effect. Pairwise comparisons did not find any significant effect of *DMT* on efficiency. *H1* was supported. Number of overshoots was significantly affected by GUIDANCE, $F(3, 262) = 6.74$, $p < .001$, and GUIDANCE × DIFFICULTY, $F(3, 262) = 4.92$, $p = .002$. Pairwise comparisons indicated that across tasks, in the *difficult* condition, participants made significantly fewer overshoots with *DMT & Token* ($M = 14.71$, $SD = 7.68$) than with *Token* ($M = 21.53$, $SD = 11.42$), $p < .001$, $r = .62$; and significantly fewer overshoots with *DMT* ($M = 16.66$, $SD = 6.31$) than *Free* ($M = 20.25$, $SD = 10.13$), $p = .024$, $r = .38$. No significant results were found for the *easy* task. *H2* was partially supported. *H3* was supported. Across the two tasks, post-hoc tests did not find any significant differences between *Free* and *Token* for either *easy* or *difficult* task. **Figure 10** shows an example of how participants' input was constrained using DMTs.

User Comments & Exploratory Task

After exploring the *Epidemic Spread* prototype, participants did not indicate significant preference toward any of the interaction modalities: *Mouse*, *Free*, *Token*, or *DMT*, $p = .166$. As expected, perceived overall workload for touchless was significantly higher than mouse, $p < .001$, $r = .77$. However, when compared with the mouse, some participants expressed a qualitative preference for using touchless interaction on large displays, for example, *"[The] mouse was restricting me to move around the large display"* [P13]. Overall, participants found touchless to be "*easy to pick up*" [P16], felt that it made them feel "*more connected*" [P15], and was simply "*more fun*" [P5].

Based on participants' open-ended summary comments, they preferred unconstrained touchless (i.e., the *Free* condition) for completing task 1, the interaction in which a strict vertical movement was required. In this context, the *lack* of constraints *"Helped me to move easily in the complex* [more dense] *matrix"* [P13], provided "*more freedom to move around*" [P3], and felt both "*free and smooth*" [P4] and "*faster and less constrained*" [P9]. However, this freedom came with a perceived cost. Participants disliked the fact that completing the task without any guidance *felt* slower and required more effort due to the lack of precision in their input: "*I'd get really far away* [from the region of interest]" [P10]; *Free* input was "*harder to control; I needed to concentrate more*" [P6]; "[I] *wasted a lot of time as I was moving away*" [P7]. These responses are notable since we found no significant quantitative differences in task completion time between *Free* and *DMT* conditions; however, they do resonate with the fact that we observed significantly fewer overshoots with DMT guidance enabled.

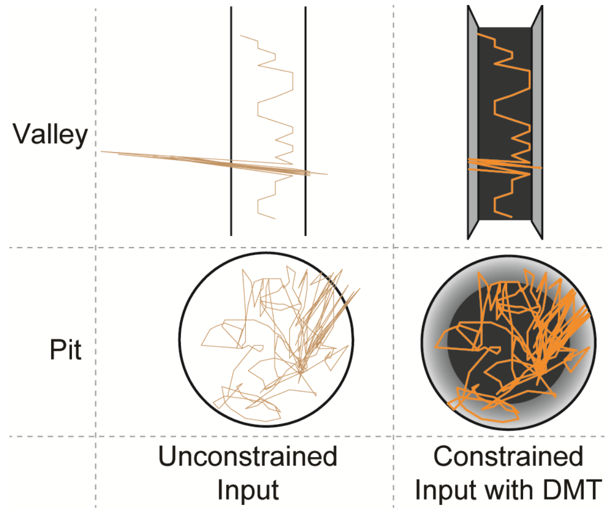

Figure 10. DMTs constrained imprecise input within the ROI: a data column (with a valley) or a data circle (with a pit).

Some participants preferred not using a physical token, because then they had "*no physical objects to use*" [P14], a perceived advantage in regards to system simplicity. They reported that *Free* touchless "*was not as accurate as token, but required less focus.*" When using the token, some participants felt guided, for example, that the token "*helped me focus*" [P2], "*felt like painting*" [P13], and "*gave me more support to move along the track*" [P3]. For these participants, the token increased confidence in input: "*I felt like I had more control*" [P6] and "*I felt I could more accurately focus my interactions*" [P16]). However, participants also reported increased fatigue when using a token and perceived the system's response to this style of input as being less precise than *Free*. For example, participants reported that "*I didn't like token. It felt less precise, less accurate*" [P4]; "*I was more tired with the token; my arm felt rigid*" [P16]; "*I expected to be more precise using the token compared with my hand. But it was not happening, so there was a break of expectations*" [P17]; and "*It felt more natural with hand, but more* straight *with token*" [P5, referring to the linear gesture needed to complete task 1].

The shortcomings of unconstrained touchless input (i.e., *Free*) that participants disliked was somewhat mitigated when using the Data-Morphed Topographies (DMT). Participants perceived the sense of cursor guidance that DMT was designed to provide: "*The subtle corrections were making me efficient*" [P16]; "*It was smoothing my… movements and keeping me in line"* [P12]; "*I did not have to continually focus on my hands afraid of getting out. It was easy to learn and helped me to be precise*" [P17]. However, some participants found the guidance to be too constraining, especially for Task 1: "*It was too much constraining in the vertical movement*" [P17]; "*I didn't like the fact that I was not in control*" [P4]; One participant reported that it was "*hard to get into another area. You don't know how much force you need to get out of the region*" [P3]. One participant [P6] found the "trail" feedback to be distracting, while some participants reported it to be useful: *"[It] lets you know that you're out of the region. You can see if you are going out with your peripheral vision*" [P4]; "*I knew when I was out of line*" [P6]

Overall, participants were divided on whether DMT was useful in task 1 or too constraining. But almost all found DMTs to be helpful in task 2. These qualitative findings are interesting, because quantitative results suggested the almost contrary: participants were significantly accurate in their touchless interaction with DMTs in task 1 compared with *Free* touchless; in task 2, however, the improvement in accuracy with DMTs only approached significance.

DISCUSSION

Overall, we found support for our main hypotheses. Data-morphed topographies (DMTs) improved the precision of touchless input (*H2*), but neither significantly affected user efficiency (*H1*) nor perceived workload (*H4*). For task 1 (**Figure 8**), which required greater overall precision than task 2, DMTs significantly increased input accuracy for the difficult trials. However, for difficult trials of task 2 (**Figure 9**), the effect of DMTs on input accuracy only approached significance. Moreover, the accuracy of easy trials in both tasks 1 and 2 was not affected by DMTs. Thus, *H3* was supported, and *H2* was partially supported. In the following section, we discuss the theoretical and practical implications of our findings.

Key Implications

Topographies improve Touchless Input Precision

Past studies provided empirical evidence that users can successfully identify macroscopic pseudo-haptic textures, such as bumps and holes, when simulated by modifying the C/D ratio of the mouse cursor [12, 13]. Building upon pseudo-haptic textures, we (1) introduced topography primitives, (2) demonstrated adaptive techniques to morph data visualizations into topographies using such primitives, and (3) provided empirical confirmation that data-morphed topographies do increase the accuracy of touchless input. Practical implications of our findings include (1) providing dynamic guidance in natural user interactions where typical haptic feedback is lacking or insufficient and (2) building *virtual affordances* for natural user interface components such as touchless menus or widgets. Notably, we found evidence that virtual constraints such as DMTs are effective only when an interaction is sufficiently difficult, i.e., for operations requiring enough input precision (H3).

Interaction Fluency vs. Input Control in Touchless Interfaces

Our findings suggested a dichotomy between users' perceived performance (as self-reported) and real performance (as auto-logged). While DMTs improved users' accuracy in both task 1 (significantly) and task 2 (approaching significance), most users reported DMTs helpful in task 2, but often too constraining in task 1 (see **Figures 8 & 9**). This tension between *interaction fluency* and *input control* is a customary tradeoff in any user interaction technique. For example, mouse allows more input control because of its characteristic resistance; but pen, touch, or touchless gestures provide more interaction fluency—a hallmark of natural user interfaces (NUI). Thus, intending to provide more user control in natural user interfaces, such as touchless, implies immediately robbing off some of its 'naturalness' or interaction fluency.

While we showed pseudo-haptic feedback improves touchless accuracy, further research is required to understand how to optimally tradeoff between interaction fluency and input control in touchless interactions—with input control mediated by feedback and user abilities [5]. To build next-generation interfaces, researchers have advocated designing feedback and affordance languages [15]. Toward generating such a vocabulary, we think a fundamental and holistic basis can be accomplished by first revisiting interaction modalities along the continua of interaction fluency and input control, then exploring mediators affecting input control, and finally placing different natural user interfaces along these dimensions.

Other Practical Implications

Apart from browsing data visualizations, DMTs can also be applied to other touchless user interfaces, such as mid-air sketching interfaces [19] or UI components (menus or widgets). For example, touchless circular menus (TCM) were found to be less accurate than linear menus with grab gestures as users were required to constrain their freehand movements between triggering a TCM and crossing-to-select a menu option [2]. The *valley* topography primitive could guide such movements and improve users' accuracy.

Limitations. Our system's performance and our study's findings are limited by the capability of our off-the-shelf tracking sensors. We evaluated DMTs using simple, abstract tasks. Further research is required to assess their performance benefits with more realistic InfoVis tasks. For example, in analytical tasks, where users' cognitive load is high, will DMTs help users to make more accurate gestures?

CONCLUSION AND FUTURE WORK

Data-Morphed Topographies improved the accuracy of users' touchless input in difficult data browsing tasks. With topographies, users made less overshoots during details-on-demand tasks than with unconstrained touchless input. We believe our contributions are threefold. First, we designed three topography primitives—holes, valleys, and pits—that map to common visualization primitives, nodes, lines, and regions. As demonstrated with additive topography, these primitives can be used to morph data visualizations to virtual topographies. Second, we proposed adaptive topographies to mitigate structural limitations of DMTs and dynamically constrain touchless input. Third, we presented results of a controlled experiment which showed that DMTs improve accuracy of touchless interactions without increasing users' overall workload or efficiency.

Our future work will investigate other applications of topography primitives, such as augmenting touchless menus or generating learning aides for dynamic touchless gestures. To build better natural user interfaces, such as touchless, also worth examining is how to optimally tradeoff between interaction fluency and input control.

ACKNOWLEDGEMENTS

REFERENCES

1. Ballendat, T., Marquardt, N., & Greenberg, S. (2010). Proxemic interaction: designing for a proximity and orientation-aware environment. *Proc. ITS*, 121–130, ACM.
2. Bates, D., and M. Maechler. 2010. Package 'lme4.' http://lme4.r-forge.r-project.org/
3. Chattopadhyay, D., & Bolchini, D. (2014). Touchless circular menus: Toward an intuitive UI for touchless interactions with large displays. *Proc. AVI*, 33–40, ACM.
4. Chattopadhyay, D., Achmiz, S., Saxena, S., Bansal, M., Bolchini, D., & Voida, S. (2014). Holes, pits, and valleys: Guiding large-display touchless interactions with data-morphed topographies. *Adjunct proc. UbiComp*, 19–22, ACM.
5. Chattopadhyay, D., & Bolchini, D. (2015). Motor-intuitive interactions based on image schemas: Aligning touchless interaction primitives with human sensorimotor abilities. *IWC*. doi: 10.1093/iwc/iwu045
6. Death of the Desktop, Envisioning Visualization without Desktop Computing (2014). Workshop co-located with IEEE VIS 2014, Paris. http://beyond.wallviz.dk/
7. Dostal, J., Hinrichs, U., Kristensson, P. O., & Quigley, A. (2014). SpiderEyes: Designing attention-and proximity-aware collaborative interfaces for wall-sized displays. *Proc. IUI*, 143–152, ACM.
8. Fung, R., Lank, E., Terry, M., & Latulipe, C. (2008). Kinematic templates: End-user tools for content-relative cursor manipulations. *Proc. UIST*, 47–56, ACM.
9. Grinstein, G., Cook, K., Havig, P., Liggett, K., Nebesh, B., Whiting, M., ... & Knoecni, S. (2011). VAST 2011 Challenge: Cyber security and epidemic. *IEEE VAST'11*, 299–301.
10. Ishak, E. W., & Feiner, S. K. (2006). Content-aware scrolling. *Proc. UIST*, 155–158, ACM.
11. Jakobsen, M. R., & Hornbæk, K. (2014). Up close and personal: Collaborative work on a high-resolution multitouch wall display. *TOCHI*, *21*(2), Article 11.

12. Lécuyer, A., Burkhardt, J. M., & Etienne, L. (2004). Feeling bumps and holes without a haptic interface: The perception of pseudo-haptic textures. *Proc. CHI*, 239–246, ACM.

13. Lécuyer, A., Burkhardt, J. M., & Tan, C. H. (2008). A study of the modification of the speed and size of the cursor for simulating pseudo-haptic bumps and holes. *ACM Transactions on Applied Perception*, *5*(3), Article 14.

14. Liu, C., Chapuis, O., Beaudouin-Lafon, M., Lecolinet, E., & Mackay, W. E. (2014). Effects of display size and navigation type on a classification task. *Proc. CHI*, 4147–4156, ACM.

15. Nancel, M., Wagner, J., Pietriga, E., Chapuis, O., & Mackay, W. (2011). Mid-air pan-and-zoom on wall-sized displays, *Proc. CHI*, 177–186, ACM.

16. Robles-De-La-Torre, G., & Hayward, V. (2001). Force can overcome object geometry in the perception of shape through active touch. *Nature*, *412*(6845), 445-448.

17. Robertson, G., Czerwinski, M., Baudisch, P., Meyers, B., Robbins, D., Smith, G., & Tan, D. (2005). The large-display user experience. *IEEE Computer Graphics and Applications, 25*(4), 44–51.

18. Sodhi, R., Glisson, M., & Poupyrev, I. (2013). AIRREAL: Tactile interactive experiences in free air. *Adjunct Proc. UIST*, 25–26, ACM.

19. Taele, P., & Hammond, T. (2014). Developing sketch recognition and interaction techniques for intelligent surfaceless sketching user interfaces. *Adjunct Proc. IUI*, 53–56, ACM.

20. Van Mensvoort, I. (2002). What you see is what you feel: Exploiting the dominance of the visual over the haptic domain to simulate force-feedback with cursor displacements. *Proc. DIS*, 345–348, ACM.

21. Wigdor, D. (2010). Architecting next-generation user interfaces. *Proc. AVI*, 16–22, ACM.

22. Yi, J. S., ah Kang, Y., Stasko, J. T., & Jacko, J. A. (2007). Toward a deeper understanding of the role of interaction in information visualization. *IEEE TVCG, 13*(6), 1224–1231