Proceedings of the ASME 2018 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2018
August 26-29, 2018, Quebec City, Canada

DETC2018 - 86141

KINESTHETICALLY AUGMENTED MID-AIR SKETCHING OF MULTI-PLANAR 3D CURVE-SOUPS

Ronak R. Mohanty
Umema H. Bohari
Vinayak*
Department of Mechanical Engineering
Texas A&M University
College Station, Texas 77843, USA
Email: ronakmohanty@tamu.edu
Email: umemabohari_123@tamu.edu
Email: vinayak@tamu.edu

Eric Ragan
Department of Visualization
Texas A&M University
College Station, Texas 77843, USA
Email: eragan@tamu.edu

ABSTRACT

We present haptics-enabled mid-air interactions for sketching collections of three-dimensional planar curves—3D curve-soups—as a means for 3D design conceptualization. Haptics-based mid-air interactions have been extensively studied for modeling of surfaces and solids. The same is not true for modeling curves; there is little work that explores spatiality, tangibility, and kinesthetics for curve modeling, as seen from the perspective of 3D sketching for conceptualization. We study pen-based mid air interactions for free-form curve input from the perspective of manual labor, controllability, and kinesthetic feedback. For this, we implemented a simple haptics-enabled workflow for users to draw and compose collections of planar curves on a force-enabled virtual canvas. We introduce a novel force-feedback metaphor for curve drawing, and investigate three novel rotation techniques within our workflow for both controlled and free-form sketching tasks.

1 Introduction

Sketching is as an essential tool for design conceptualization. It embodies a certain *controlled vagueness* [1] that enables users to quickly externalize their ideas without a need for a finished product. To date, most successful digital sketching workflows are predominantly implemented using multi-touch interactions on tablets for the simple reason that they offer a close resemblance to the traditional pen-and-paper medium, thus significantly reducing the interface learning curve for a novice user.

With recent advances in augmented and virtual reality and computer vision technologies, there is a significant interest in expanding the scope of sketching from 2D media to 3D spaces [2]. At the same time, it is also argued that sketching 2D representations of 3D ideas adds to cognitive load, especially for novice designers leading to sketch inhibition [3]. While this view is echoed by existing interaction design research [4–6] as well, little is understood regarding the underlying principles of interaction design for mid-air sketching interfaces. Our goal, in this work, was to explore interactive techniques to enable a kinesthetically augmented experience for 3D sketching during conceptual design.

Inspired by existing approaches [7–9] for tablet-based 3D sketching, we introduce a novel interaction workflow for 3D sketching that allows users to sketch 3D curves through a 6 Degree of Freedom(DoF) haptic stylus. Our intention, however, was not to develop a feature-rich system, but to study the spatiality and tangibility of mid-air interactions for 3D sketching tasks through our simple and succinct workflow. For this, we developed a novel force-feedback approach to allow users to draw spatial curves, and studied three novel techniques for rotation of curves in 3D space.

*Address all correspondence to this author.

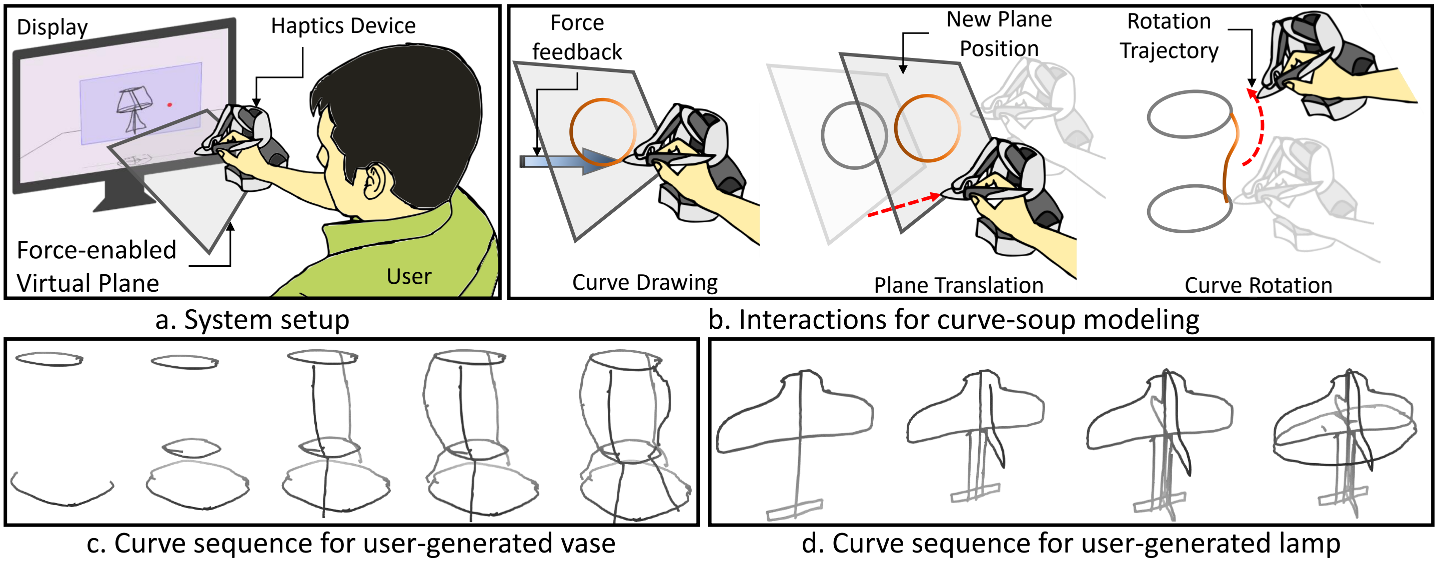

FIGURE 1. General overview of the creative workflow and design representation: User draws using a haptic device on a virtual plane in mid-air, which is rendered on the screen (a), Interaction workflow comprises of (C) *curve drawing* , (T) *translation* of sketch plane, and (R) 3D *curve rotation* (b), User creates a multi-planar 3D curve-soup of a vase and a lamp using C,R,T (c)(d).

The evaluation of these techniques offer some interesting insight into the challenges of enabling paper-like sketching experiences in mid-air interactions.

In this paper, we make two main contributions. First, we demonstrate a simple 3D sketching workflow that: (a) builds on existing spatial interactions and haptics approaches to create novel kinesthetic experiences for drawing in mid-air and (b) preserves controlled vagueness of sketching as an expressive medium for constraint-free externalization of ideas. Second, we conduct a formal user-based evaluation of the underlying interaction techniques for 3D curve drawing, translation, and rotation to better understand (a) the role of mid-air interactions in effectively utilizing physical movement in 3D sketching, (b) how different interaction strategies affect user perception, experience, and performance in 3D sketching tasks, and (c) how kinesthetic feedback can be used to augment the process of conceptualization in mid-air.

2 Related Work

Our work draws from several known approaches for interactive approaches for 3D sketching, mid-air interactions and haptics. Below, we motivate and contrast our work with respect to relevant and related works.

There are two categories of approaches that address the creation of 3D sketches. The first category deals with the process of creating these sketches using tablet-based multi-touch interactions. In their work *ILoveSketch*, Bae et al. [8] introduced a comprehensive system for expert designers to create refined 3D sketches for conceptualization. They further extended their approach to cater to novice users in their system *EveryodyLovesSketch* [9] using simplified interactions. However, both these systems ultimately produce fine quality sketches that naturally take time to create and do require training. In contrast, *MentalCanvas* by Dorsey et al. [7] allows for quicker creation of multi-planar curves that are more reminiscent of actual rough sketches. Tsang et. al [10] demonstrated interactions to allow users to draw 3D sketches using a 2D image as a guide in the background. Recently, Xu et al. [11] proposed a projection technique that converted user drawn 2D sketches to 3D curve networks. The second class of 3D sketching research focuses on 3D user input [12–14]. One of the early works in this area is the *3-Draw* system by Sachs et al. [15]. Xin et al. [16] introduced *NapkinSketch*, a novel system for drawing multi-planar sketches in 3D space in a tablet-based see-through augmented reality environment.

The fundamental issue occurring with bare hand gesture based mid-air interactions is that while they can be very effective for short interactions (such as object selection), they lack the tangibility and kinesthetic control necessary for involved tasks such as concept sketching [17]. Although novice users find 3D interactions for sketching as intuitive and free, lack of tangibility and depth perception adds to user's cognitive load [18]. Several works have addressed this through with specialized hardware controllers [19] or haptics-enabled interactions [20]. Schkolne et al. [21] suggest a 3D drawing system that use hand motions and tangible tools for sketching and manipulation of 3D curves. Keefe et al. [20] demonstrated a haptics enabled bi-manual interactive system controlled creation of 3D line illustrations in a virtual environment. Here, the idea was to constrain user movement by providing a resistive force-feedback during the drawing of a free-form curve using a haptic device [22]. Haptics devices have also been employed for continuous indirect 3D object manipulation [23] in virtual sculpting tasks in CAVE virtual reality systems. It was observed that stereoscopic vision wasn't alone sufficient for depth perception and required some tangible reaction force in the form of haptics. Raymaekers et. al [24] demonstrated haptics enabled sketching system for creation and modification of 3D curves using cubic Beziér splines. Similarly, Fünfzig et al. [25] allowed the user to edit curves using haptics while maintaining its qualitative attributes.

2.1 Our Contributions

Our aim was to provide users with a virtual mid-air drawing canvas that preserved the tangibility of physical pen-and-paper sketching experience. Prior works by Oakley et al. [26] and Miller et al. [27] have explored haptics on 2D GUIs for interactive purposes. Taking inspiration from work done by Raymaekers et. al [24], we propose multi-planar modeling of curves.

Our work is different from past works in two major ways. First, our interaction workflow is intended as a direct spatial extension to how one would produce a sketch on paper. Unlike earlier works, this extension is in the process of sketching (3D user input), the outcome of sketching (the 3D sketch), and the experience of sketching (perception of a piece of paper, only floating in 3D space). Second, our work adds to the existing body of work on 3D manipulation [28, 29] in the context of curve modeling by decomposing the DoF for curve creation and manipulation respectively.

3 Interaction Design

In this research work, we envision our curve-soups as a spatial collection of planar curves residing on different 3D planes, and configured relative to each other so as to provide an abstract visual representation of 3D solid objects (Figures 1(c), (d)).

3.1 System Setup

Our system (Figure 1(a)) comprises of a computer screen showing a virtual plane that acts as a canvas for users to draw planar curves, and a GeoMagic Touch 6 DoF haptics device. The idea is to allow the user to use the stylus of the device as a pen to draw planar curves in space. The stylus is equipped with two push-buttons that we employed to implement our interaction workflow.

3.2 Design Rationale

To enable curve-soup modeling, the fundamental requirement is that users should be able to draw planar curves in 3D space,

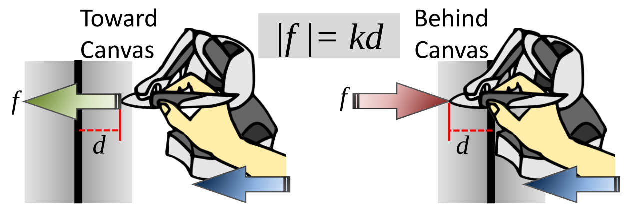

FIGURE 2. Algorithm for proximity based kinesthetic feedback for detection of drawing plane in mid-air.

at any desired relative positions and orientations with respect to each other. From the point of view of input, our setup affords only two capabilities: positioning and orienting the haptic stylus and the button press on the stylus. Drawing from the work by Jacob et al. [30], we identify that the process of creating 3D curve collections can be naturally segmented into three fundamental operations: drawing, rotation, and translation. Below, we expand on how we use our input capabilities to enable these operations.

3.3 Interaction Workflow

There are three fundamental operations that compose our interaction workflow: (a) curve drawing, (b) canvas translation, and (c) curve-soup rotation as described below.

3.3.1 Curve Drawing

We designed a haptics-enabled interaction wherein, given a fixed plane in the virtual environment, users could draw curves on the plane while experiencing a force-feedback against the stylus in the users' hand. A simple way to achieve this would be to simply provide a reaction force along the plane's normal. However, our initial pilot studies showed that the lack of depth perception in a predefined zone around the plane senses the presence of the stylus and activates the creation mode. This also activates the haptic feedback for sketching and the user feels a springing force based on the stylus movement in the Z direction. The user can pull in and out of the plane or move sideways to exit the sketching zone. Primarily, the user can start and stop sketching by staying out the boundary of the sketch plane.

3.3.2 Canvas Translation

We considered two options for allowing positional control of the existing set of curves: (a) direct translation of curves in the scene and (b) translation of the virtual canvas. We found the latter to be a better approach since it minimized the confusion caused by the global movements of the entire curve-soup in the scene. Also, it minimized accidental input strokes by users trying to translate the curves. To put canvas translation into effect, a user would press and drag the back button provided on the stylus. Here, the canvas was rigidly affixed to the stylus tip allowing for direct and proportional translation.

An additional option we considered was to provide users

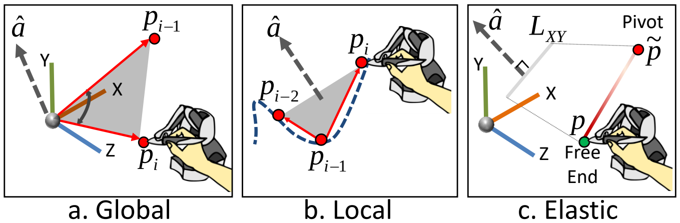

FIGURE 3. Axis and angle are computed about the Global (G) origin (a), Axis and angle are computed using Local (L) stylus trajectory (b), and Axis and angle related to Elastic (E) length of the line about a fixed pivot (c).

with orientation control of the canvas by aligning its normal to the orientation of the stylus. However, we constrained the canvas to be parallel to the front (X-Y) plane of the global coordinate system. The primary reason for this was the limited range-of-motion of the haptic manipulator that restricted user's ability to sketch on arbitrarily oriented planes in our pilot experiments. Also, incoherency due to user input on an arbitrarily oriented plane and its visual feedback on the display added to user's cognitive load.

3.3.3 Curve-soup Rotation

In order to allow users to draw curves at desired relative orientations to each other, we intended to design an interaction wherein the user could rotate all existing curves in the scene at once about the global origin using the stylus trajectory. However, choice of a single mid-air rotation scheme posed a challenge. In mouse or touch based interactions [31, 32], where the axis and angle of rotation is inferred through widgets such as arc-ball [33]) or composed multi-touch inputs on the screen [34]. These ray-casting approaches have been extended to mid-air interactions by Katzakis et al. [35] in their *Mesh-Grab* and *Arc-Ball 3D* approaches. While these would work well for surfaces and solids, applying the same to curve is challenging merely by the attribute of the thickness (or the lack thereof) of curves and the resulting lack of precise controllability. Thus, we designed three novel approaches for curve-soup rotation. In all cases, the idea is to compute an axis of rotation residing in the global X-Y plane and an angle of rotation based on the 3D stylus trajectory.

G Global Rotation: Given two consecutive points, $\mathbf{p}_i$ and $\mathbf{p}_{i-1}$, on the trajectory, the axis is computed as the normalized cross-product $\hat{\mathbf{a}} = \hat{\mathbf{p}}_{i-1} \times \hat{\mathbf{p}}_i$ and the angle is computed as $\theta = \arccos(\hat{\mathbf{p}}_{i-1} \cdot \hat{\mathbf{p}}_i)$ (Figure 3(a)). This is, in spirit, similar to *Arc-Ball3D* proposed by Katzakis et al. [35].

L Local Rotation: In this case, we used differential geometry to compute the axis and the angle. We consider the triangle formed by three consecutive points, $\mathbf{p}_i$, $\mathbf{p}_{i-1}$, and $\mathbf{p}_{i-2}$, on the stylus trajectory. Subsequently, the axis is computed as the signed normal to the plane defined by this triangle, i.e. $\hat{\mathbf{a}} = \hat{\mathbf{v}}_1 \times \hat{\mathbf{v}}_2$ where, $\mathbf{v}_1 = \mathbf{p}_{i-2} - \mathbf{p}_{i-1}$ and $\mathbf{v}_2 = \mathbf{p}_i - \mathbf{p}_{i-1}$ (Figure 3(b)). In this case, we define the angle as $\theta = c\|\mathbf{p}_i - \mathbf{p}_{i-2}\|$. We determined the constant c through trial and error through pilot experiments.

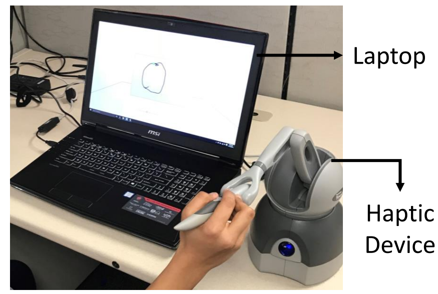

FIGURE 4. Hardware setup comprising of display (laptop) and a haptic device.

E Elastic Rotation: Unlike mouse or touch based interactions, the lack of physical support leads to significant physical labor in mid-air interactions. In our third technique, our aim was to devise a rotation approach for reducing arm movement while enabling users to rotate objects faster. For this, we designed an indirect rotation approach [34] wherein the stylus trajectory was mapped to the velocity of rotation instead of providing incremental angular changes. In this approach, the press of the stylus button at a given point, $\tilde{\mathbf{p}}$, fixes this point in space as a pivot point. For any subsequent stylus point $\mathbf{p}$, we orthogonally project the line $L(\mathbf{p}, \tilde{\mathbf{p}})$ on the X-Y plane. We then compute the axis $\hat{\mathbf{a}}$ such that $\hat{\mathbf{a}} \perp L_{XY}(\mathbf{p}, \tilde{\mathbf{p}})$. Instead of directly computing the angle of rotation, we compute the angular velocity $\omega = b\|\mathbf{p}\tilde{\mathbf{p}}\|$. This effectively gives the user an illusion of stretching an elastic string to rotate the curves with variable speeds based on the amount of stretch (Figure 3(c)).

4 Implementation

4.1 Hardware & Software

Our hardware (Figure. 4) comprises of a MSI Dominator GT72 laptop computer with Intel Core i7-6700HQ CPU (3.6GHz, 16GB GDDR5 RAM), running 64 bit Windows 10 Professional with a NVIDIA GeForce GTX 1070M graphics card (8GB video memory). Our curve-soup modeling application was developed in C++ with openGL Shading Language for rendering.

4.2 Transitioning across Operations

To provide a smooth transition across drawing, translation, and rotation, we used the two buttons provided in the haptic stylus. To rotate the curve-soup and translate the drawing canvas, a user would press-and-drag the front and the back buttons respectively. Here, *dragging* simply means moving in 3D space along an arbitrary path. For drawing on the canvas, the user would simply go in the proximity of the virtual canvas within a pre-defined threshold. Here, our plane-snapping method (Figure 2) would provide force-feedback during drawing. The users experience a springing effect confirming activation of the creation mode aided by color coded visual cues. The force-feedback algorithm directs the user towards the plane from both directions along the normal of the plane. However, while sketching, the user experiences a smooth constant force (f) along the virtual plane.

4.3 Stylus Trajectory Smoothing

As the force-feedback algorithm tends to snap the stylus to the virtual sketching plane, we observed users experiencing a jerking movement causing unintentional curve inputs to be recorded on the plane. In order to avoid jerky curve input while drawing or manipulating the curves, we apply a low-pass filter to the stylus trajectory by using exponential smoothing [36, 37]. Given a point $\mathbf{v_t}(x_t, y_t, z_t)$ of the stylus trajectory at an instance t, we compute the smoothened coordinates $\hat{\mathbf{v}}_\mathbf{t}(\hat{x}_t, \hat{y}_t, \hat{z}_t)$ as:

$$\hat{\mathbf{v}}_\mathbf{t} = \alpha \mathbf{v_t} + (1-\alpha)\mathbf{v_{t-1}} \quad (1)$$

Here, $\alpha \in [0,1]$ is the smoothing coefficient. We apply this process to all stylus trajectories across all three interfaces.

4.4 Interface Refinement

Prior to a formal evaluation of our 3D curve drawing system, we wanted to first understand the general usage patterns of users, weaknesses of our interface and users' reaction to the fundamental components of the interface (especially planar force-feedback while sketching). For this preliminary study, 11 participants used our interface and performed tasks like generating wire-frame models of primitives and creating free-form shapes. Based on user feedback, we made the following improvements to our interface:

Visual Cues : Shadows of the stylus tip, sketch plane and sketched 3D curves, were added for better depth perception. In addition, depth based color gradient from black to gray was provided to the sketched 3D curves.
Haptics : Force value for plane-snapping caused random strokes to be drawn when user entered or exited the sketch zone. This was due to the jerk caused by high force value while in proximity of the sketch plane. We reduce the force value by changing k = 0.5. This minimized the impact of force while sketching, but maintaining the awareness created by a smoother plane-snapping.
Device Workspace Mapping: We observed haptic device workspace limiting user movement while reaching the extremes of the interface. Thus, the workspace was remapped to the openGL workspace allowing full movement of the haptic stylus while sketching in 3D and preventing any kind of gimbal lock.

5 Evaluation

We conducted a user study to evaluate the three rotation techniques (*Global* **G**, *Local* **L**, *Elastic* **E**) based on curve drawing accuracy, rotation efforts, and controllability.

5.1 Participants

The participants group involved a mix of 18 (5 female, 13 male) students (23 - 30 years old) from engineering, architecture, and visualization majors. Based on the demographics survey, the participants belonged to two categories: *experts* with extensive experience with 3D sketching and 3D modeling software, and *novices* having limited or no experience with either 3D sketching or modeling.

5.2 Evaluation Tasks

Our evaluation was designed with three goals in mind. First was to individually evaluate force-feedback for curve drawing and canvas translation. Secondly, we wanted to compare the three rotation techniques introduced in our interface with respect to user performance, preferences, and behavior for sketching 3D curve soups. Finally, we also wanted to observe the use of our overall interaction workflow in a design conceptualization scenario through 3D sketching. Based on these goals, we designed the following evaluation tasks for users to perform.

T1 *Mesh-guided modeling*: In this task, participants were shown a 3D mesh of a primitive shape on the interface and were asked to draw a curve-soup representing the edges of the model.The curve-soup would visually resemble a wireframe model of the 3D mesh. This was meant to allow us to quantitatively assess user performance in terms of shape accuracy and completion time for each of our rotation techniques. We chose a cube and a frustum (truncated pyramid) as our shapes as both objects have well-defined sharp features (edges at adjacent faces). The idea was to use these features to provide users with cues to draw their curves so as to match these features. Also, the difference in shape uniformity provided insight on a user's perception of depth during the sketching tasks. While we did not have a rigid time duration for this task, we controlled each trial for this task between 5 and 7 minutes.

T2 *Open-ended modeling*: Our goal was to (a) understand how our overall workflow would be used in a typical design scenario without a volumetric visual cue (as in **T1**) and (b) identify unique usage patterns with respect to our rotation techniques. Each participant was asked to conceptualize an object from familiar product contexts as a 3D sketch. Here, we wanted users to create objects that contained smooth features (unlike **T1**), could be reasonably depicted using non-linear planar curves (such as lofts and sweeps), and allowed users flexibility to apply their creative interpretation as in a conceptual design process. For this, we chose product contexts such as vases and lamp-shades in this task.

Both the tasks (**T1** and **T2**) were performed for all three rotation techniques (**G**, **L**, and **E**) in a randomized order using Latin square across participants to account for learning biases.

5.3 Procedure

The experiment took approximately 60 minutes. Each session started with the general introduction of the haptic device and user interface, familiarizing the participants with the interface and interaction workflow (drawing, rotation, and translation) for creating a curve-soup. This was followed by an initial demographic questionnaire. The experiment subsequently consisted of the following tasks:

Practice : Participants began by creating a simple set of curves (such as a wire-frame model of a cube, a tree etc) for 5 minutes. We ensured that they used all three interaction modes during practice.

T1 : Each participant performed three trials for creating a 3D curve-soup for a cube followed by the frustum using all three rotation techniques.Thus, a total of 6 trials were performed per participant. After each rotation technique, we recorded participants feedback using the NASA task-load index [38]. Participants were encouraged to use the rotation, translation, undo, and redo operations for accessing different faces.

T2 : Each participant created either a curve-soup representing a lamp-shade or a vase using all three rotation techniques. Thus, performing 3 trials per participant. No references in the form of mesh models were provided, and participants were encouraged to use the interactions for sketching creative shapes. This task lasted 5–7 minutes. Subsequently, each participant responded to a questionnaire regarding the general interface features, use of haptic feedback during drawing, and a combined comparison of the three rotation techniques. We also collected open-ended comments regarding the overall interaction workflow.

5.4 Data & Metrics

For each trial performed by a participant, we recorded (a) the raw event log containing time-stamped stylus trajectory for each OpenGL frame along with the current mode (i.e., drawing, rotation, translation, undo, and redo), (b) the final 3D curve-soup, (c) user feedback, and (d) live video of the participant.

For the mesh-guided tasks (**T1**), our goal was to quantify the deviation of the user generated curve soup with respect to the ground-truth, i.e., the edges on the cube and frustum models. While there are existing approaches in sketch based accuracy evaluation such as the one proposed by Badam et al. [39], they measure performance in pixel space. In contrast, our techniques are in geometric space. Thus, we used the root mean squared (RMS) error for this quantification as follows:

$$E_{RMS} = \sqrt{\frac{\sum_{i=1}^{n} d_i^2}{n}} \tag{2}$$

Here, d_i is the distance between a point p_i in the user drawn curve and the edge of the target model (cube or frustum).

6 Results

In the following sections, we report on the statistical analysis of our three rotation techniques and discuss the main insights we gained from our data collection, observation, and user feedback from 162 total trials performed by all participants.

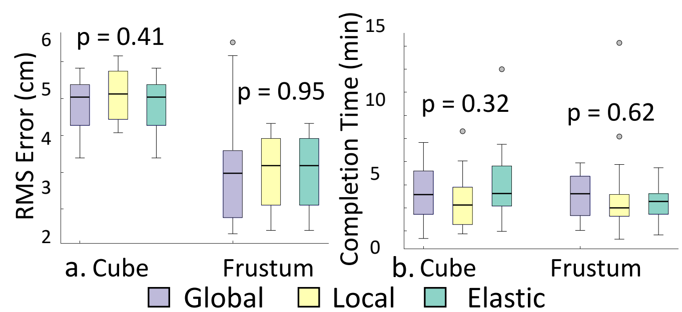

FIGURE 5. (a) RMS Error for rotation techniques for Cube and Frustum, (b) Completion time for rotation techniques for Cube and Frustum (b).

6.1 User Performance

6.1.1 Drawing Accuracy (T1):

We make the following hypotheses here:

Null (Ho): The mean RMS for each mesh type is equal across all rotation variants.

Alternate (Ha): The mean RMS error for elastic variant is lower than other two rotation variants.

We first verified the normality of the RMS error data per rotation technique for each mesh type in **T1** using Shapiro-Wilk test. Following which, we compared both mesh types for all three rotation variants using one-way ANOVA test. The p-values for each comparison were above the significance level ($\alpha = 0.05$), validating our null hypothesis **Ho**. This suggests that despite different manipulation techniques, users could trace along meshes with same accuracy for all rotation variants.

On comparing cube and frustum mesh types,the mean RMS error was observed to be higher in the former for all three rotation variants(Figure 5 (a)). The global variant exhibited relatively lower RMS error for both mesh types, but the range of error values range was relatively higher in case of a frustum mesh. We believe that the relative non-orthogonality of the faces of the frustum resulted in the users not being able to align its face to the sketch plane.

6.1.2 Completion Time (T1):

Despite of allotting a fixed time, the participants were allowed to complete the task until they felt satisfied with their output. Thus, we observed a variation of completion times across the tasks (Figure 5 (b)). We make the following hypotheses here:

Null (Ho): The mean completion time for each mesh type is equal across all rotation variants.

Alternate (Ha): The mean completion time for elastic variant is lower than other two rotation variants.

We first verified the normality similar to RMS error FOR completion time data using Shapiro-Wilk test, and showcased comparison across each mesh type for all three rotation variants using one-way ANOVA test. The p-values for each comparison were above the significance level ($\alpha = 0.05$), validating our null hypothesis **Ho**. This suggests that despite different manipulation techniques, users completed the tracing tasks no sooner than other rotation variants.

Our belief for a relatively higher mean completion time in the elastic variant is due to its indirect nature of rotation interaction with the curves.Thus, causing users to spend more than usual time in tracing accurately.

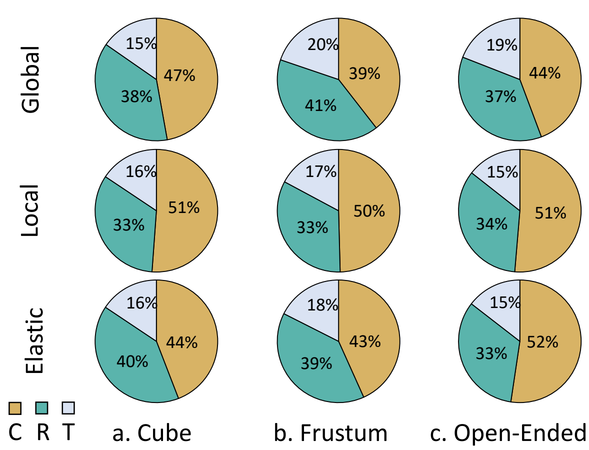

FIGURE 6. Percentage time distribution of events : *curve drawing* (C), *translation* (T), and *rotation* (R) events across Global, Local and Elastic techniques for Cube, Frustum and Open-ended tasks.

6.1.3 Time Distribution Across Operations (T1 & T2)

We make interesting observations relating rotation-techniques and type of tasks in terms of amount of labor required to manipulate the curve-soup. This is supported by our observation for task completion times (Figure 5 (c), (d)). On analyzing percentage time distribution per interaction mode (Figure 6) for a cube, maximum rotational effort was required for elastic rotation technique, followed by global and local rotation techniques. Thus, for the cube, task completion is relatively quicker for local rotation technique with minimum effort, and relatively more for elastic technique with maximum effort. We can relate the increased physical effort being mapped to increase in sketch accuracy of the elastic technique. In case of frustum, maximum rotation effort is reported for global rotation technique, followed by elastic and local techniques. However, an inverse relation between accuracy and completion times is observed in this case. Thus, unlike the cube mesh, physical effort of users is not mapped to sketch accuracy for a frustum. We can observe that local technique remains invariant to the complexity of the task. For open-ended curve-soup modeling tasks, completion time increases with increase rotation efforts. Elastic(3.6 minutes) required the least effort and global(5 minutes) required the most among the three techniques.

6.2 User Feedback & Observations

We collected open-ended feedback on completion of sketching tasks for every interface. We discuss some relevant feedback in conjunction with our own observations during the tasks.

6.2.1 Interaction Workflow

Kinesthetic feedback during mid-air sketching for all the tasks was taken positively by the users. Users were also able to translate the sketching plane with minimum effort using the haptic stylus. However, a few users were dissatisfied with the lack of orientation control for the virtual plane. One user stated: "*Is it possible to rotate the plane*?". We observed the reason as users with prior experience in 3D sketching and 3D modeling trying to co-relate the rotation techniques to the ones available on commercial 3D modeling softwares. This explains the relatively large dispersion in error for global and elastic techniques in frustum and cube tasks respectively.

6.2.2 Rotation Techniques

For rotation, users found it easier to relate with the global and local techniques, owing to their direct manipulation nature. However, elastic wasn't welcomed with much appreciation due to its indirect nature. Eventually, users did prefer the visual cue for making fine rotations. One user intuitively mentioned, "*I can make finely control rotation to make detailed sketches to my design, which I am unable to do in Solidworks or any other CAD software .*" An interesting observation in open-ended tasks showed users having different approach methods based on the rotation technique for creating

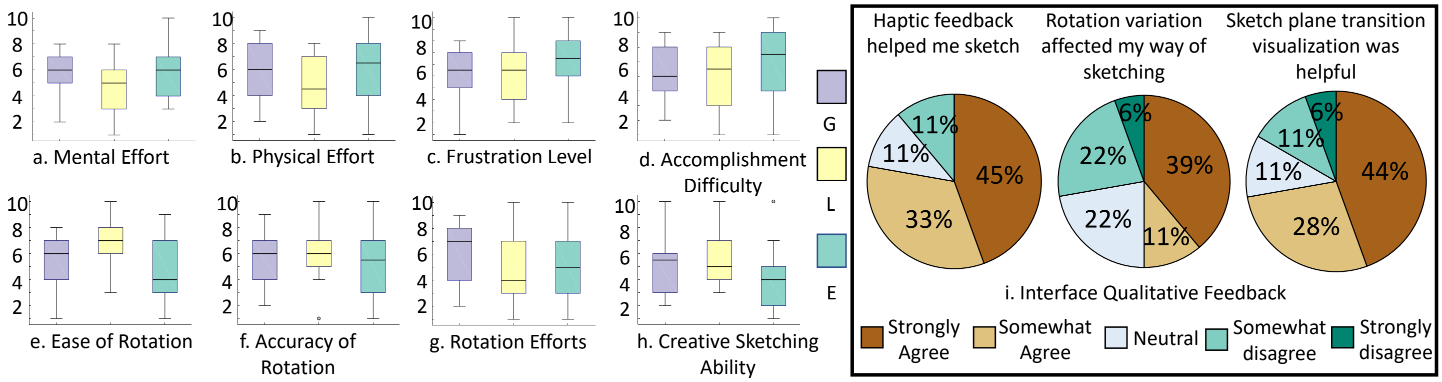

FIGURE 7. (a), (b), (c), (d) [1:low; 10:high]: Qualitative feedback for the individual rotation techniques. (e)[1:difficult; 10:easy], (f) [1:least; 10:most accurate], (g)[1:high; 10:low effort], (h) [1:difficult; 10:easy]: User comparison between rotation techniques.

the same curve-soup model across all interfaces.

The participants felt that the mental and physical effort required for rotation was found to be least for the local technique. Whereas, the global technique seemed a less frustrating choice for users to rotate with lesser difficulty (Figure 7(a), (b), (d)).

We observed that the order of the techniques affected users' perception of the techniques. For instance, users who began the trials with elastic technique faced an initial struggle due to the technique's indirect nature for rotation manipulation adding to their frustration (Figure 7(c)). It is interesting to note that even though users agreed that the elastic technique involved less physical effort, they gave higher importance to the directness of the interaction offered by the global approach.

The local technique was rated relatively easier by users on the basis of ease of rotation (Figure 7(e)). However, global technique was found to involve least efforts while rotating, which also aided the user's creative ability during the open-ended task (Figure 7(g), (h)). The overall rotation accuracy ratings were similar across all three interfaces.

6.2.3 Force-feedback for Drawing

As expected, users found it difficult to acclimatize to the snapping metaphor for the rendering force on the virtual plane. They were specifically frustrated while entering the sketching zone from either direction. This, however, reduced with practice over time. Interestingly, we found that our force-feedback method also provided subtle depth cues to users while drawing curves. One user said "*Force feedback is a good indicator. It allowed me to sense where the sketch plane was in 3D space*." While most participants could easily adapt to this interaction, a few participants (primarily from the *expert* category) mentioned the need for an explicit start-stop button. One user stated: *"I wish there was an extra button for drawing, it becomes difficult to switch between drawing and rotating or translating the plane"*. Users made explicit use of *undo* and *redo* commands to refine their sketches as they went ahead with the tasks.

6.2.4 Visual Cues & Depth Perception

Users made a considerable usage of visual cues provided by our interface across all three rotation techniques. Overall, users were positive about shadows aiding as a depth cue while drawing multi-planar curves in 3D. Most users hinted at the presence of mesh models aiding their depth cues during the initial tasks. For the open-ended task, where no reference mesh model was provided, one user mentioned, *"Pretty easy to draw stuff with visual feedback. Everything becomes very difficult when there is no reference"*. Although, shadows were provided for the sketch plane too, users found it difficult to translate and sketch a new planar curve in 3D. Another user mentioned, *"Shadows helped, but needed something else to understand where the sketching plane was."* For the open-ended task (**T2**), both *novice* and *expert* participants were successfully able to sketch detailed wire-models of either a vase or a lamp shade, effectively using the depth cues (Figure 8).

6.3 Observations

Across 18 participants, 9 trials were performed per user i.e. 3 tasks per variant per task. Overall, all participants were positive about the general idea of being able to create rough 3D concepts using multi-planar curves. Most users were comfortable with using the haptic device for translation and rotation interactions with the amount of practice that was allowed.

General Perception : We observed an interesting diversity in perception of 3D across all users. Due to the dearth of 3D sketching applications in commercial use, it was difficult for them to correlate 3D modeling with our tasks, for creating 3D sketches. However, initial practice and additional information gave these participants and insight into our concept and they were comfortable in sketching in 3D. For our novice participants, those with an introductory exposure to 3D modeling and sketching, it was relatively easier for them to decipher depth in 3D compared to

FIGURE 8. Curve-soup models for lamp-shade and vases created by users during the open-ended task

experts. There were a few with no background in 3D modeling and they faced difficulty correlating planar curve creation to overall creation of a curve-soup. Surprisingly, one user devised an implicit method of immediately switching to plane transition using the back button on the stylus, in order to avoid the haptic feedback of sketch zone.

Fatigue: It was expected for users to face fatigue due to prolonged mid-air interaction (about 45 minutes) for the trial tasks. However, force feedback during sketching compensated for the mid-air hanging of user's hand. As most users were involved in performing the tasks, very few user explicitly complained about hand fatigue. We asked users to rest their hands in between the tasks for user comfort and better productivity of their sketches.

Limitations: The planar nature of the sketch plane was found to be a matter of concern across few users. This was due to their intent of creating non-planar curves in 3D. However, the inherent problem of workspace limitation of the haptic device restricts user movement beyond a certain degree of freedom. As currently, there are no untethered devices providing haptic feedback, this is a design trade off in our interface due to enable creation of curve-soup models. Also, lack of sufficient visual cues for depth perception added to user difficulty while sketching in 3D. Our interface lacked selective deletion of curves, because of which, user's couldn't delete a particular curve without deleting the ones succeeding it.

7 Discussion

7.1 Perceived Utility

We found users to be very agreeing with our original goal — preserving the essence of traditional sketching in mid-air interactions for design conceptualization. Especially, expert participants mentioned the potential for it to replace 2D sketches for design ideation. One user stated,"*As designers, we begin with concept generation using 2D sketches on a paper. Your system could allow us to ideate directly in 3D, providing more details about our design concepts.*" Another user mentioned, "*I am looking forward to such an application in future, making idea generation quicker and easier for us*."

7.2 Kinesthetic Feedback for Curve Input

There is a rich space of unexplored kinesthetic interactions that are yet to be investigated for 3D sketching. For instance, extending the plane snapping approach to 3D curves would enable perform close-range operations such as curve refinement, deformation, and over-sketching [40] — operations that are common on tablets but extremely difficult in mid-air. Furthermore, such snapping would allow one to create topologically connected *curve-networks*, as opposed to curve-soups. Such curve-networks could be subsequently used for generating surface models [41].

7.3 The Best rotation technique?

While we intended to find one best approach for rotations, our evaluations showed a clear value for both global and elastic approaches; the first clearly performed well in terms of user preference while the second in terms of reducing physical labor. This is unlike the case of Katzakis et al. [35] where *Arc-Ball3D* was a clear winner. We believe this is due to the fact that our rotation was in context of a design conceptualization task where rotation was merely one operation in an interactive sequence targeted toward a broader design task rather than for manipulation alone.

7.4 Composition of rotation techniques

Continuing on the previous point, our evaluations indicate that there is room for combining multiple approaches, especially global and elastic, for 3D mid-air sketching to offer on-demand priority toward reduced physical labor (in case of elastic) or better precision (as offered by global). Decision regarding which to use when could be based on factors such as complexity of the curve-soup, frequency of re-orientation, and user fatigue. Automatic switching between different rotation modes is an interesting future research direction.

8 Future Directions and Conclusions

We explored a kinesthetic based approach for augmenting mid-air sketching based curve-soups. As a core contribution, we provided an interaction workflow for direct 3D design conceptualization. Our goal in the immediate future is to perform a quantitative and qualitative evaluation of the rotation techniques using haptics for 3D sketch manipulation. In particular, we want to understand how user perception and performance changes for manipulation tasks with and without haptic feedback. We will also study how experience, performance, and design outcomes changed with addition of haptics for manipulation. Finally, it will be interesting to see how we could extend our interaction workflow into a full fledged 3D sketching system for designers and novice users to create 3D curve models.

ACKNOWLEDGMENT

This work was supported by the Texas A&M University start-up fund, AggiE_Challenge at Texas A&M University, and the Department of Mechanical Engineering at Texas A&M University.

REFERENCES

[1] Stappers, P., and Hennessey, J., 1999. "Toward electronic napkins and beermats: Computer support for visual ideation skills". In *Visual Representations and Interpretations*, R. Paton and I. Neilson, eds. Springer London, pp. 220–225.

[2] Wikipedia contributors, 2018. Tilt brush — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Tilt_Brush&oldid=842029842`. [Online; accessed 30-May-2018].

[3] Jonson, B., 2005. "Design ideation: the conceptual sketch in the digital age". *Design Studies*(6), pp. 613–624.

[4] Zeleznik, R. C., Forsberg, A. S., and Strauss, P. S., 1997. "Two pointer input for 3d interaction". In Proceedings of the 1997 Symposium on Interactive 3D Graphics, I3D '97, ACM, pp. 115–120.

[5] Zeleznik, R., and Forsberg, A., 1999. "Unicam—2d gestural camera controls for 3d environments". In Proceedings of the 1999 Symposium on Interactive 3D Graphics, I3D '99, ACM, pp. 169–173.

[6] Smith, G., Salzman, T., and Stuerzlinger, W., 2001. "3d scene manipulation with 2d devices and constraints". In Proceedings of Graphics Interface 2001, GI '01, Canadian Information Processing Society, pp. 135–142.

[7] Dorsey, J., Xu, S., Smedresman, G., Rushmeier, H., and McMillan, L., 2007. "The mental canvas: A tool for conceptual architectural design and analysis". In Proceedings of the 15th Pacific Conference on Computer Graphics and Applications, PG '07, IEEE Computer Society, pp. 201–210.

[8] Bae, S.-H., Balakrishnan, R., and Singh, K., 2008. "Ilovesketch: As-natural-as-possible sketching system for creating 3d curve models". In Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology, UIST '08, ACM, pp. 151–160.

[9] Bae, S.-H., Balakrishnan, R., and Singh, K., 2009. "Everybodylovessketch: 3d sketching for a broader audience". In Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology, UIST '09, ACM, pp. 59–68.

[10] Tsang, S., Balakrishnan, R., Singh, K., and Ranjan, A., 2004. "A suggestive interface for image guided 3d sketching". In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '04, ACM, pp. 591–598.

[11] Xu, B., Chang, W., Sheffer, A., Bousseau, A., McCrae, J., and Singh, K., 2014. "True2form: 3d curve networks from 2d sketches via selective regularization". *ACM Trans. Graph.,* ***33***(4), July, pp. 131:1–131:13.

[12] Deering, M. F., 1995. "Holosketch: a virtual reality sketching/animation tool". *ACM Transactions on Computer-Human Interactions,* ***2***(3), pp. 220–238.

[13] Keefe, D. F., Feliz, D. A., Moscovich, T., Laidlaw, D. H., and LaViola, Jr., J. J., 2001. "Cavepainting: a fully immersive 3d artistic medium and interactive experience". In Proceedings of the ACM symposium on Interactive 3D graphics, pp. 85–93.

[14] LaViola, J. J., and Keefe, D. F., 2011. "3d spatial interaction: Applications for art, design, and science". In ACM SIGGRAPH 2011 Courses, pp. 1:1–1:75.

[15] Sachs, E., Roberts, A., and Stoops, D., 1991. "3draww: A tool for designing 3d shapes". *IEEE Comput. Graph. Appl.,* ***11***(6), Nov., pp. 18–26.

[16] Xin, M., Sharlin, E., and Sousa, M. C., 2008. "Napkin sketch: Handheld mixed reality 3d sketching". In Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology, VRST '08, ACM, pp. 223–226.

[17] Arora, R., Kazi, R. H., Anderson, F., Grossman, T., Singh, K., and Fitzmaurice, G., 2017. "Experimental evaluation of sketching on surfaces in vr". In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, CHI '17, ACM, pp. 5643–5654.

[18] Israel, J., Wiese, E., Mateescu, M., Zllner, C., and Stark, R., 2009. "Investigating three-dimensional sketching for early conceptual designresults from expert discussions and user studies". *Computers & Graphics,* ***33***(4), pp. 462 – 473.

[19] Balakrishnan, R., Fitzmaurice, G., Kurtenbach, G., and Buxton, W., 1999. "Digital tape drawing". In Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology, UIST '99, ACM, pp. 161–169.

[20] Keefe, D. F., Zeleznik, R. C., and Laidlaw, D. H., 2007. "Drawing on air: Input techniques for controlled 3d line illustration". *IEEE Transactions on Visualization and Computer Graphics,* ***13***(5), pp. 1067–1081.

[21] Schkolne, S., Pruett, M., and Schröder, P., 2001. "Surface drawing: creating organic 3d shapes with the hand and tangible tools". In Proceedings of the ACM conference on Human factors in computing systems, pp. 261–268.

[22] Massie, T., 1998. "A tangible goal for 3d modeling". *IEEE*

Comput. Graph. Appl., ***18***(3), May, pp. 62–65.

[23] Foskey, M., Otaduy, M. A., and Lin, M. C., 2002. “Artnova: Touch-enabled 3d model design”. In Virtual Reality, 2002. Proceedings. IEEE, IEEE, pp. 119–126.

[24] Raymaekers, C., Vansichem, G., and Van Reeth, F., 2002. “Improving Sketching by Utilizing Haptic Feedback”.

[25] Fnfzig, C., Thomin, P., and Albrecht, G., 2010. “Haptic manipulation of rational parametric planar cubics using shape constraints”. In Proceedings of the ACM Symposium on Applied Computing, pp. 1253–1257.

[26] Oakley, I., McGee, M. R., Brewster, S., and Gray, P., 2000. “Putting the feel in ’look and feel‘”. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI ’00, ACM, pp. 415–422.

[27] Miller, T., and Zelevnik, R., 1998. “An insidious haptic invasion: Adding force to the X desktop”. In Proceedings of the 11th annual ACM symposium on User interface software and technology, ACM, pp. 59–66.

[28] Veit, M., Capobianco, A., and Bechmann, D., 2009. “Influence of degrees of freedom’s manipulation on performances during orientation tasks in virtual reality environments”. In Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology, VRST ’09, ACM, pp. 51–58.

[29] Veit, M., Capobianco, A., and Bechmann, D., 2010. “Dynamic decomposition and integration of degrees of freedom for 3-d positioning”. In Proceedings of the 17th ACM Symposium on Virtual Reality Software and Technology, VRST ’10, ACM, pp. 131–134.

[30] Jacob, R. J. K., Sibert, L. E., McFarlane, D. C., and Mullen, Jr., M. P., 1994. “Integrality and separability of input devices”. *ACM Trans. Comput.-Hum. Interact.*, pp. 3–26.

[31] Hand, C., 1997. “A survey of 3d interaction techniques”. *Computer Graphics Forum*, pp. 269–281.

[32] Bowman, D. A., Kruijff, E., LaViola, J. J., and Poupyrev, I., 2004. *3D User Interfaces: Theory and Practice*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA.

[33] Shoemake, K., 1992. “Arcball: A user interface for specifying three-dimensional orientation using a mouse”. In Proceedings of the Conference on Graphics Interface ’92, Morgan Kaufmann Publishers Inc., pp. 151–156.

[34] Knoedel, S., and Hachet, M., 2011. “Multi-touch rst in 2d and 3d spaces: Studying the impact of directness on user performance”. In Proceedings of the 2011 IEEE Symposium on 3D User Interfaces, 3DUI ’11, IEEE Computer Society, pp. 75–78.

[35] Katzakis, N., Seki, K., Kiyokawa, K., and Takemura, H., 2013. “Mesh-grab and arcball-3d: Ray-based 6-dof object manipulation”. In Proceedings of the 11th Asia Pacific Conference on Computer Human Interaction, APCHI ’13, ACM, pp. 129–136.

[36] Holt, C. C., 2004. “Forecasting seasonals and trends by exponentially weighted moving averages”. *International Journal of Forecasting,* ***20***(1), pp. 5 – 10.

[37] Thiel, Y., Singh, K., and Balakrishnan, R., 2011. “Elasticurves: Exploiting stroke dynamics and inertia for the real-time neatening of sketched 2d curves”. In Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, UIST ’11, ACM, pp. 383–392.

[38] Hart, S. G., and Staveland, L. E., 1988. “Development of nasa-tlx (task load index): Results of empirical and theoretical research”. *Advances in psychology,* ***52***, pp. 139–183.

[39] Badam, S. K., Chandrasegaran, S., Elmqvist, N., and Ramani, K., 2014. “Tracing and sketching performance using blunt-tipped styli on direct-touch tablets”. In Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces, AVI ’14, ACM, pp. 193–200.

[40] Baudel, T., 1994. “A mark-based interaction paradigm for free-hand drawing”. In Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology, UIST ’94, ACM, pp. 185–192.

[41] Orbay, G., and Kara, L. B., 2012. “Sketch-based surface design using malleable curve networks”. *Computers & Graphics*, pp. 916 – 929. Graphics Interaction Virtual Environments and Applications 2012.

