Proto-TAI++: Exploring perceptually consistent creation of planar shape assemblies through multi-modal tangible interactions

Cecil Piya[*], **Vinayak, Karthik Ramani**[†]
C-Design Lab
School of Mechanical Engineering
Purdue University, West Lafayette, Indiana
Email: {cpiya,fvinayak,ramani}@purdue.edu

In this work, we explore the utility of tangible 3D interactions that allow for geometric and perceptual correspondence between a mid-air modality and the 3D elements it controls. To demonstrate our approach, we use a concrete application scenario through Proto-TAI++, a multi-modal system that allows for pen-based drawing of planar shapes and their subsequent mid-air assembly using a hand-held planar shaped proxy. The planarity of the proxy is a key element that physically embodies virtual planar shapes during 3D manipulation. We also investigate advanced operations such as shape patterning and blending by exploiting asymmetric bimanual interactions that augment mid-air motion with multi-touch gestures. We describe a three-stage user evaluation with our system wherein our goal is to (a) study the effectiveness of the planar proxy as a tangible 3D modality, (b) contrast key features of our approach with a GUI-based planar shape assembly system, and (c) evaluate user experience and performance in creative tasks using the Proto-TAI++ system. A video description of this work can be found in ***https://youtu.be/Kwi2M81ijwQ***

1 Introduction

Holding and manipulating physical objects is a natural task that humans learn to perform at an early age. Drawing inspiration from this, several works, most notably *graspable user interfaces* [2], have explored tangible interactions where hand-held manipulation of physical objects are used for expressively controlling virtual elements. Ishii [3] stated that within such interfaces, "*the physical forms serve as both representations and controls for their digital counterparts,*", and Fitzmaurice et al. [2] point out that "*the affordances of the physical handles are inherently richer than what virtual handles afford through conventional (desktop) manipulation techniques.*". In general, the notion of using physical objects as media for virtual manipulation has been found particularly relevant to digital 3D interactions [4].

While several works have utilized generic 3D input devices as tangible media for virtual manipulation, we find that there is little correspondence between the device and the virtual 3D shapes being manipulated, in terms of the geometric form of the objects and that of the device. Recently, Arisandi et al. [5] demonstrated *Virtual Handcrafting* as an interesting example of directly using the real world tool metaphors for virtual modeling. Inspired by such works, we aim to explore the use of physical objects that bear geometric and semantic similarity with their virtual counterparts. Here, the premise is that given such correspondence, users can apply physical interactions on the objects to achieve perceptually consistent outcomes in the virtual space. To this end, we explore using ordinary objects (re-purposed from our surroundings) as physical embodiments of virtual 3D elements, for direct 3D manipulations of such elements.

To investigate the utility of perceptual consistency in tangible interactions, we take an application oriented approach by choosing the context of planar shape assembly. This context serves two purposes. First, given the planarity of the basic modeling elements, they lend themselves for spatial manipulations using a planar shaped object. This object serves as a tangible proxy for holding, manipulating, and assembling virtual planar shapes in 3D space. We posit that the resulting interactions can allow users to vicariously configure planar shapes using suggestive mid-air actions that are consistent with the intended outcome. Second, the simplicity of this context allows us to study the proposed interactions in-depth, and to explore their utility within creative design scenarios.

Planar shape abstractions have found interest in art and design, mainly for representing 3D ideas in a minimalistic form, while using simple fabrication techniques like laser cutting. Their utility in digital 3D modeling has also been shown in various creative activities like early-stage design ideation, digital art, and fabrication of functional artifacts. Our work mainly deals with interactive construction of such

[*] Address all correspondence to this author.
[†] School of Electrical Engineering (by courtesy)

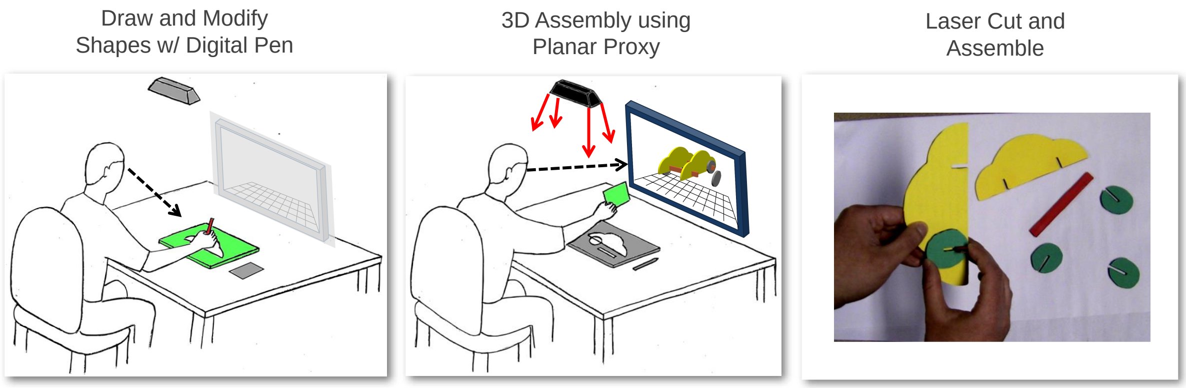

Fig. 1. The general workflow of *Proto-TAI++*: (a) planar shapes are drawn and modified on a 2D interface with a digital pen, (b) mid-air inputs from the planar proxy and concurrent multi-touch inputs on the tablet (with non-dominant hand) are used for coordinating assembly operations in the 3D modeling space, (c) the final planar shape assembly can be fabricated with a laser cutter and physically assembled [1]. The following link provides a demo of the system usage and supporting interactions **https://youtu.be/Kwi2M81ijwQ**

assemblies in a virtual 3D setting. Prior works in this area [6, 7] have primarily utilized digital pen-based drawing with subsequent fabrication and physical assembly of planar shapes. However, we find that this approach poses a gap between shape creation and assembly realization, resulting in extraneous iteration cycles and waste of resources. Thus, in our earlier work [1] we demonstrated *Proto-TAI*, a system that integrated virtual 3D assembly with 2D drawing within a common interface, allowing users to iterate and explore designs within the virtual space itself. The idea here was to demonstrate the utility of a planar shaped proxy for metaphorically possessing and configuring planar shapes in mid-air.

Recently, we have observed additional interest in the research community towards interactive planar shape assembly. Most notably, works like *FlatFitFab* [8] have demonstrated how advanced modeling capabilities (e.g. procedural operations, interactive shape modification etc.) can significantly enhance the aesthetic quality and level of detail in planar shape assemblies. Motivated with these developments, we present our extended system , *Proto-TAI++*, and explore the utility of our interactions towards supporting such advanced modeling capabilities. In addition, given the robustness and accessibility of *FlatFitFab*, it serves as a basis for evaluating our interactions in context of planar shape assembly, and comparing key differences between the two approaches.

1.1 Contributions

Our first contribution is *Proto-TAI++*, a multi-modal system for creating planar shape assemblies using both multi-touch and tangible mid-air interactions. This system extends our earlier work [1] by including operations such as interactive planar shape modification (Section 5.1 (b)), automatic assembly adjustment (Section 5.2.3), and procedural operations (Section 5.2.2). While our goal is to significantly improve the complexity and level-of-detail within the models constructed, we also aim to retain the simplicity of the interactions found in our earlier work. We achieve this in two ways. First, we explore a comprehensive asymmetric bimanual interaction scheme, where tangible mid-air inputs with the planar proxy are coupled with touch based event specification (Figure 14 (a-f)). We then extend these interactions towards more precise and structured modeling operations by contextually constraining free-form mid-air inputs (Figure 14 (g-i)).

Our second contribution is the in-depth evaluation of our interactions as well as the *Proto-TAI++* system through three user studies. The first study evaluates the efficacy of the planar proxy for supporting both free-form and constrained manipulation of virtual shapes (Section 4). In the second study, we compare the fundamental interactions and key workflow differences between our approach and *FlatFitFab* (Section 6.1). We present the results in terms of task performance, system feature usability, and user experience. Finally, we study various creative outcomes that can be achieved using the final *Proto-TAI++* system (Section 6.2).

2 Related Works

3D Design with Planar Shapes Planar shapes have been commonly used in CAD to define sweep sections of 3D extrusions. Other interactive modeling systems have also utilized spatial arrangement of sketched profiles [9] or multi-view image silhouettes [10] as seed geometry in organic 3D design. While these methods infer 3D form from planar shapes, other works present automatic algorithms to extract planar shape abstractions form existing 3D models [11–13].

Our work in contrast involves interactive creation and assembly of planar shapes into meaningful 3D structures from a blank slate. Sketch-It-Make-It [7] and Designosaur [6] provide a similar mode of design expression, where users can draw planar shapes on a digital sketch medium for subsequent fabrication and assembly. Similarly, Interactive Construction [14] allows users to provide laser-based sketch inputs to guide real-time fabrication of planar shapes. But since these methods do not provide a digital modeling space for 3D assembly, users cannot explore different assembly configurations and inter-shape compatibility before fabrication. This can increase the number of *ideate-construct-iterate* cycles, leading to loss of time and material. In contrast, FlatFitFab [8] enables a collocated virtual drawing and assembly medium, along with procedural operations to add complexity and details. SketchChair [15] also utilizes procedural operations to generate complex planar shape designs from simple interactions.

Our approach enables both interactive construction and assembly of planar shapes within a common digital interface. However, we use a multi-modal system where drawing and assembly operations are performed in separate 2D and 3D modeling spaces respectively. We also explore the use of a planar proxy as a mid-air modality that serves as a physical representation of virtual planar shapes during 3D assembly.

Combined 2D and Mid-Air Interfaces With advancements in both touch-sensitive media and depth sensors, several works have explored "continuous interaction spaces" [16, 17] that integrate multi-touch tabletop interactions with above-the-surface mid-air gestures. Such interfaces have been shown to allow seamless transition between 2D and 3D input spaces based on the needs of the current task. In Mockup Builder [18], the authors demonstrated the utility of such interfaces within a 3D design scenario.

Given that our workflow comprises of both 2D sketching and 3D assembly tasks, we find value in utilizing a similar dual mode interface to support fluid interactions. However, in contrast to most existing works, which utilize mid-air hand gestures, we utilize tangible mid-air interactions with a physical proxy. Since the planar proxy is non-instrumented, we also utilize concurrent asymmetrical bimanual inputs [19], where the dominant hand communicates spatial inputs in mid-air and the non-dominant hand indicates discrete events through multi-touch gestures.

Mid-Air Interactions for 3D Design Several works have explored mid-air interactions in 3D design. In contrast to GUI based 3D widgets, they enable a more direct and efficient means for providing spatial inputs, but at a coarser level of precision. We find that given the quick-and-dirty nature of planar shape assembly, this is a reasonable trade-off. Mid-air interactions are broadly classifiable as follows.

Free-hand Gestures have been increasingly explored for 3D interactions, particularly with recent developments in low-cost depth sensors. They have been utilized in various 3D design tasks like constrained mechanical and scene assembly [20] and conceptual design [21]. However, while several works have explored continuous hand orientation and pose tracking [22], its use within robust unimanual 3D object manipulation has remained a challenge [23]. In addition, free-hand gestures also lack a means to provide tactile feedback for kinesthetic control of virtual objects.

Digital Controllers provide greater spatial control and precision using in-built motion sensors and collocated click buttons for secondary inputs. Thus, various works have utilized commercial hand-held [24] or hand-worn [25] controllers for 3D design applications that require more precision. Such controllers utilize generic hardware and metaphors for interacting with a wide range of 3D objects. Given their growing ubiquity, other works [26] have also repurposed smartphones as multi-modal hardware in 3D design.

Customized Controllers have been demonstrated as task specific hardware within design contexts like virtual clay sculpting [27] and virtual handcrafting [5]. Such controllers typically repurpose ordinary objects as 3D interactive media by embedding electronic sensors or AR markers. The planar proxy in our work also serves a customized controller for mid-air interactions. By using vision-based tracking, we keep the proxy free of digital instrumentation and constructible from simple cardboard.

Planar Shaped Controllers have been used for 3D interactions in different contexts. For example, Hinckley et al. [4] used a digitally instrumented planar tool for 3D data visualization. Within 3D design, Kato et al. [28] explored a planar shaped paddle, tracked using AR markers, as a generic tool for holding and manipulating 3D models. The key difference in our work is that, we focus on using the planar proxy as a physical embodiment of modeling elements within planar shape assembly. While prior works have mainly focused on spatially configuring existing models, we also explore the use of a planar proxy in constructing new geometries and exploring creative ideas.

3 System Implementation

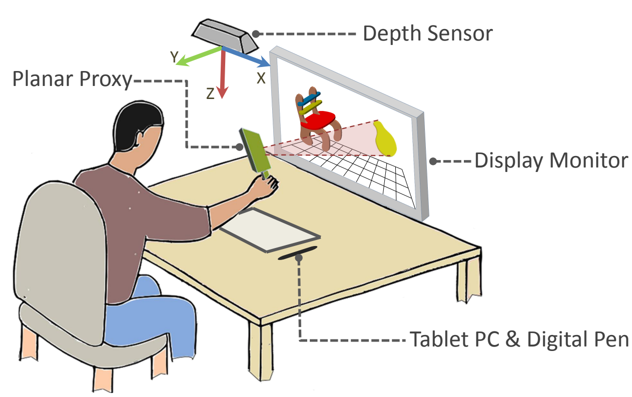

Fig. 2. Generalized setup of proposed system.

The generalized setup (Figure 2) for *Proto-TAI++* consists of (a) a tablet PC for computational power, 2D drawing, and multi-touch inputs, (b) an overhead depth sensor to track the proxy's motion, and (c) a display monitor to render the virtual 3D scene. The mid-air interaction space is defined as a 600x440x400 mm volume, lying directly over the desk surface and measured from the sensor's frame of reference. The 3D scene spatially corresponds to the interaction space, has a 14x7x8 unit volume, and is rendered in perspective view.

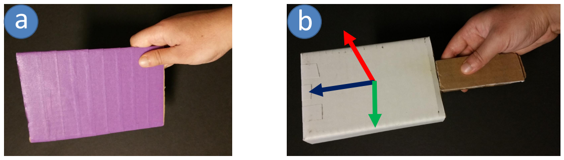

Fig. 3. Planar proxy: (a) preliminary version, (b) refined version with increased thickness and a flat handle. Arrows define local reference.

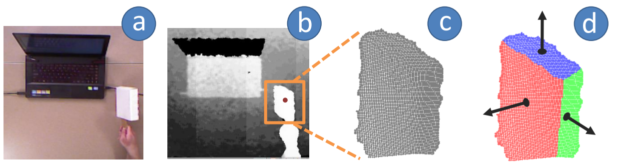

Fig. 4. Tracking proxy's motion: (a) Interaction space viewed from depth sensor, (b) Depth map with 3D position and bounding box. (c) 3D data inside bounding box, (d) Segmented faces with normals.

These dimensions can be modified based on other requirements or constraints. Initially, we had used a rectangular cardboard cutout as the planar proxy (Figure 3(a)). However, we included a flat handle on for ergonomic grasping and to allow subtle 3D inputs via wrist and fingers movements [29] (Figure 3(b)). We also increased its thickness to prevent occlusions when facing sensor along thin edge.

3.1 Tracking Proxy's Mid-air Motion

To track the planar proxy's 3D motion, we utilized a Kinect depth sensor (Figure 4). However, other vision-based approaches with fiducial markers and a web-cam are also viable [28]. The proxy's hand-held motion is tracked using the hand-tracking function in OpenNI's NITE library. The tracked position approximately lies at the proxy's center (Figure 4(b)), and is measured with respect to the sensor's frame of reference. The proxy's orientation is represented by three orthogonal directions. Our system applies the following steps to track the orientation.

Proxy Data Acquisition An axis-aligned 3D bounding box is drawn with its center at the proxy's 3D position. The point cloud data within this box is stored as the proxy data. We empirically found a 200x200x170 mm volume bounding box to be suitable for a proxy with a 100x150x25mm dimensions.

Point Cloud Segmentation Laplacian smoothing is applied on the proxy data to reduce noise, and normals computed at each point. Using spherical k-means [30] on the normals, the data is then segmented into three (or less) planar regions. Adjoining planar regions within 15 degrees are merged together. Each planar region represents a proxy face, and its normal is computed using Principal Component Analysis.

Orientation Estimation When a principle direction is unmeasurable due to occlusions, it is inferred from the other directions. For example, when one direction is missing (Figure 5 (middle)), it is set as the cross product of the other two. Similarly, two missing directions can be estimated using PCA on the single visible face (Figure 5 (right)).

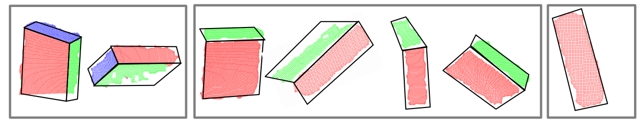

Fig. 5. Segmenting planar regions on 3D data: (left) three faces visible to sensor, (middle) two faces visible, (right) only one face visible.

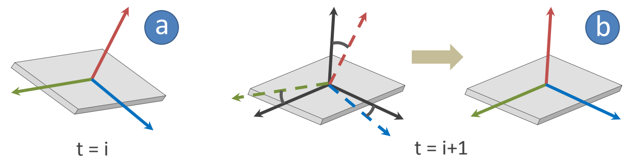

Fig. 6. Temporal coherence of orientation: (a) Face normals at frame i, (b) Assigning measured directions in frame $i+1$ to appropriate faces, based on angular proximity with normals in frame i.

Temporal Coherence of Orientation Tracking is initialized by holding the proxy, such that it is roughly aligned with the global frame of reference (top face horizontal, front face away from user). This allows the system to assign principle directions to correct faces on the proxy. At ensuing frames, a measured direction is matched to a face if its normal in the prior frame is closest to the direction. (Figure 6).

We apply single exponential smoothing (alpha = 0.3) to remove jitter during position tracking. For the orientation, its principal directions are first represented as quaternions (with respect to the global x-axis) and smoothened using the double exponential function [31].

3.2 Mapping Proxy's Motion onto Planar Cursor

Users can express *intent to interact* with the 3D scene by raising the proxy towards the sensor [32]. This activates an interactive planar cursor in the scene, which can be controlled by manipulating the proxy in mid-air (Figure 7(a)). To stop interacting, the proxy is placed back on the desk.

We impose spatial correspondence between the interaction and modeling spaces by having the sensor face vertically downwards and its X-Y axes roughly along the desk edges. This allows us to linearly map the proxy's motion onto the cursor. Here, a 5 mm linear motion of the proxy gets mapped as a 0.12 unit cursor displacement. The proxy's orientation can be directly applied to the cursor. For a different sensor orientation, the angular offset between its frame of reference and the desk surface must be accounted for.

Users can clutch a planar shape by first hovering the cursor over it, and then using a single tap gesture on the tablet (with non-dominant hand). A clutched shape gets overlaid on the cursor and moves with the proxy (Figure 7(a)). This interaction can also be generalized for controlling non-planar shapes (Figure 7(b)), except here the cursor is attached to the

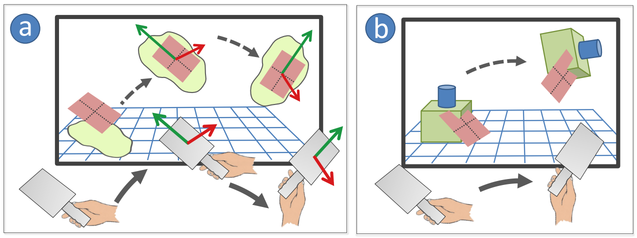

Fig. 7. Using planar cursor to clutch and manipulate: (a) a planar shape, (b) a non-planar shape.

shape across its geometry.

4 Evaluation of Planar Proxy

We conducted a study to assess proxy's utility during 3D interactions, in context of both planar shape and general 3D shape manipulations. To show the interaction's accessibility, we recruited 15 participants (11 male, 4 female) from engineering, natural sciences, and liberal arts. Here, 5 participants were familiar with 3D shape modeling, but none had prior experience with mid-air interactions. In each session, they were first familiarized with the interactions (15 minutes) and assigned the following tasks.

4.1 Task 1: Docking 3D Objects

This task evaluated spatial accuracy attainable with the proxy when configuring shapes at 3D locations. In a given trial, an asymmetrical shape was picked up and aligned with a fixed target (Figure 8 (left)). A trial was successful if the shape's position and orientation was brought within a predefine tolerance. We used 7 different shapes in each run, and tightened the tolerance between successive runs. By predefining a shape's starting and docking locations, we ensured adequate challenge, and also minimized variability during analysis. However, we randomized the sequence in which the shapes were presented to avoid learning effects.

The results indicated that users were unable to dock shapes if the positional and orientation tolerances were less than 0.05 units (equivalent of 2.1 mm in interaction space) and 2 degrees respectively. Above these values, we observed 97.8% of the trials to be completed in under 30 seconds (Figure 8 (right)), and the remaining over 45 seconds.

4.2 Task 2: Hoop through 3D Wire

This task determined if the proxy can provide reasonable control of shapes along a constrained path. The objective was to guide a circular hoop along a 3D wire without touch the wire (Figure 9 (left)). Here, the hoop was directly overlaid on the cursor, giving the impression of holding its rim. We used five trials in the study with differently shaped wires. We empirically found that a hoop with 0.5 unit inner diameter provided the right balance between task feasibility and challenge adequacy. The hoop was considered to touch the wire if (a) the distance between its center and the wire exceeded the inner hoop radius, and/or (b) the angle between the its

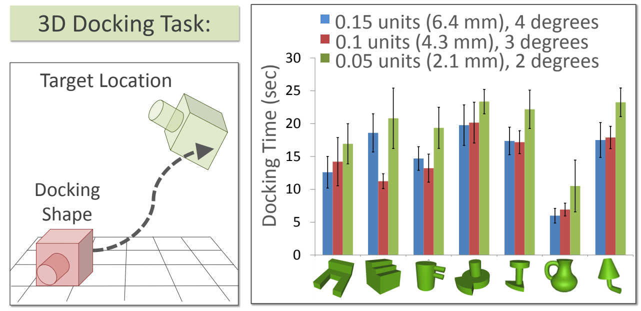

Fig. 8. Task 1: (left) docking task, (right) task completion times for 7 asymmetric shapes at three levels of docking tolerances.

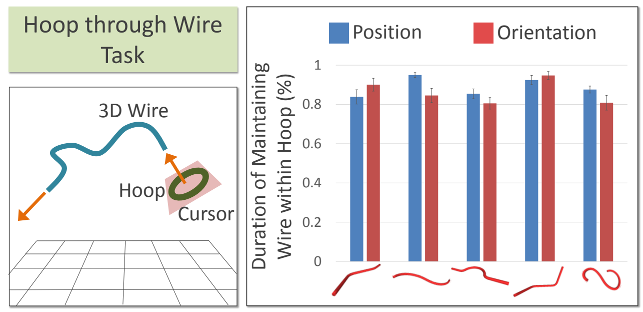

Fig. 9. Task 2: (left) guiding a circular hoop across a 3D wire, (right) duration of maintaining the wire within the hoop.

normal and the wire exceeded 35 degrees. Color changes in the wire was used to notify users of such deviations, allowing them to readjust the hoop.

For each trial we recorded the frequency of both positional and orientation deviations. Figure 9(right) shows that on average users were able to keep the hoop within both thresholds during at least 80% of the path. By observing the two deviations separately, we intended to find if one was more dominant than the other. However, the results show that the frequency of the two kinds of deviations were similar.

4.2.1 Takeaways

Here, we demonstrated the proxy's utility in supporting both 6 DOF and constrained manipulation of 3D shapes. These properties are essential during planar shape assembly, as 6 DOF manipulability enables direct configuration of shapes, while controlled manipulations allow for subtle changes in the assembly for exploring different forms.

5 Proto-TAI++

In our setup (Figure 2), planar shapes are drawn directly on the tablet using a digital pen, and subsequently assembled via mid-air interactions with the planar proxy. The following sections describe the interactions used during these two modes.

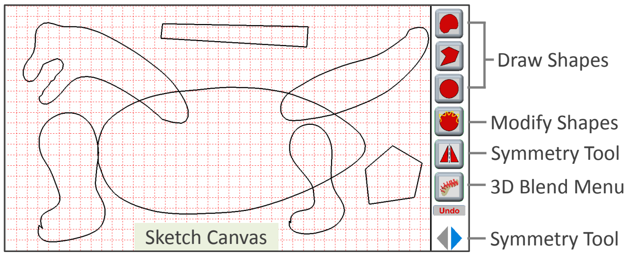

Fig. 10. 2D drawing interface displayed on the tablet.

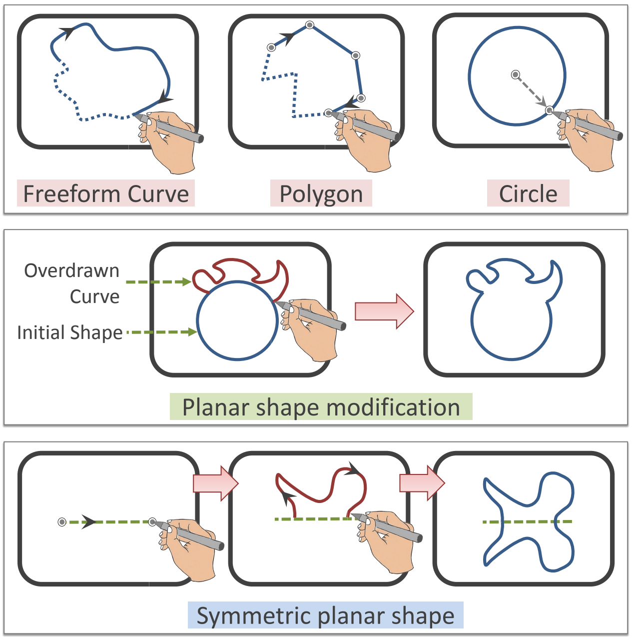

Fig. 11. Drawing operations: (top) Drawing shapes, (middle) Editing shapes, (bottom) Creating a symmetric shapes.

5.1 Drawing Planar Shapes

Figure 11 shows the drawing interface. It contains a 2D canvas with grid-lines and a set of buttons for indicating different sketch operations. All of these operations are carried out with the digital pen and are as follows.

Draw Free-form curves and circles are drawn using a single stroke input, while polygons defined through their vertices (Figure 11 (top)). Each shape is represented as a closed sequence of 2D points, which are evenly spaced using uniform re-sampling. Free-form curves are neatened using method described in [33]. Open curves are assumed to be closed by a straight line joining their end points.

Modify The overdrawing technique [34] allows users to modify a shape's geometry, add details, or improve its appearance (Figure 11 (middle)). It provides an intuitive mode for editing shapes and also reflects how most people modify drawings using pen-and-paper.

Mirror To create a symmetric shape, a line-of-symmetry is first defined. An open curve or polygon, drawn on one side of this line gets mirrored across to create a complete symmetric shape (Figure 11 (bottom)).

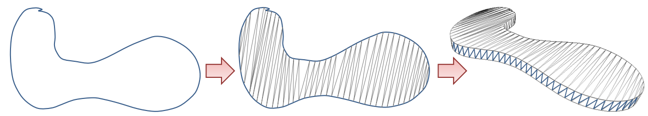

Fig. 12. Converting 2D drawing to planar mesh for use in assembly.

Shape Manipulation Shapes drawn on the canvas can be moved around or resized for directly observing and adjusting relative proportions between shapes. Similarly, by using copy, modify, and delete operations, users can create variations of one shape and explore different forms.

Generate Planar Section Each drawn shape can be saved as a planar mesh for use in the 3D assembly (Figure 12). Here, constrained Delaunay triangulation first creates a 2D mesh over the shape. The mesh is then extruded by offsetting its own copy by a pre-defined width, and stitching the two copies with a series of triangles.

5.2 3D Assembly of Planar Shapes

The 3D modeling scene (Figure 13) comprises of a horizontal desk over which the saved planar shapes are laid out. To avoid clutter, only 6 shapes are displayed at a time, but the scroll buttons allow access to other shapes. Shadows are rendered on the desk surface to provide depth perception. The trash-bin is used for discarding unwanted assembly shapes.

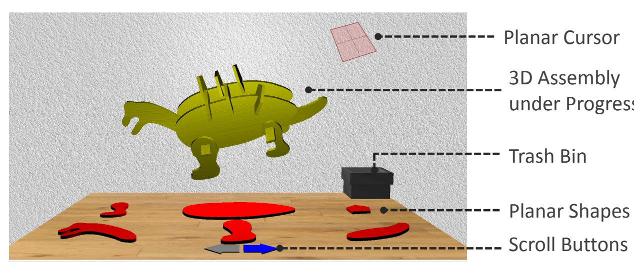

Fig. 13. 3D scene for assembling planar shapes.

Here, 3D interactions are performed by coordinating the cursor's hover state (dominant hand) with multi-touch inputs (non-dominant hand). Touch gestures can be applied anywhere on the tablet's surface, precluding the need to directly look at the tablet. Visual feedback as color changes indicates proximity between the cursor and the scene elements. Figure 14 illustrates assembly operations. Here, each cell shows the proxy's motion in the interaction space, the cursor's hover state in the 3D scene, the touch input on the tablet, and their collective effect.

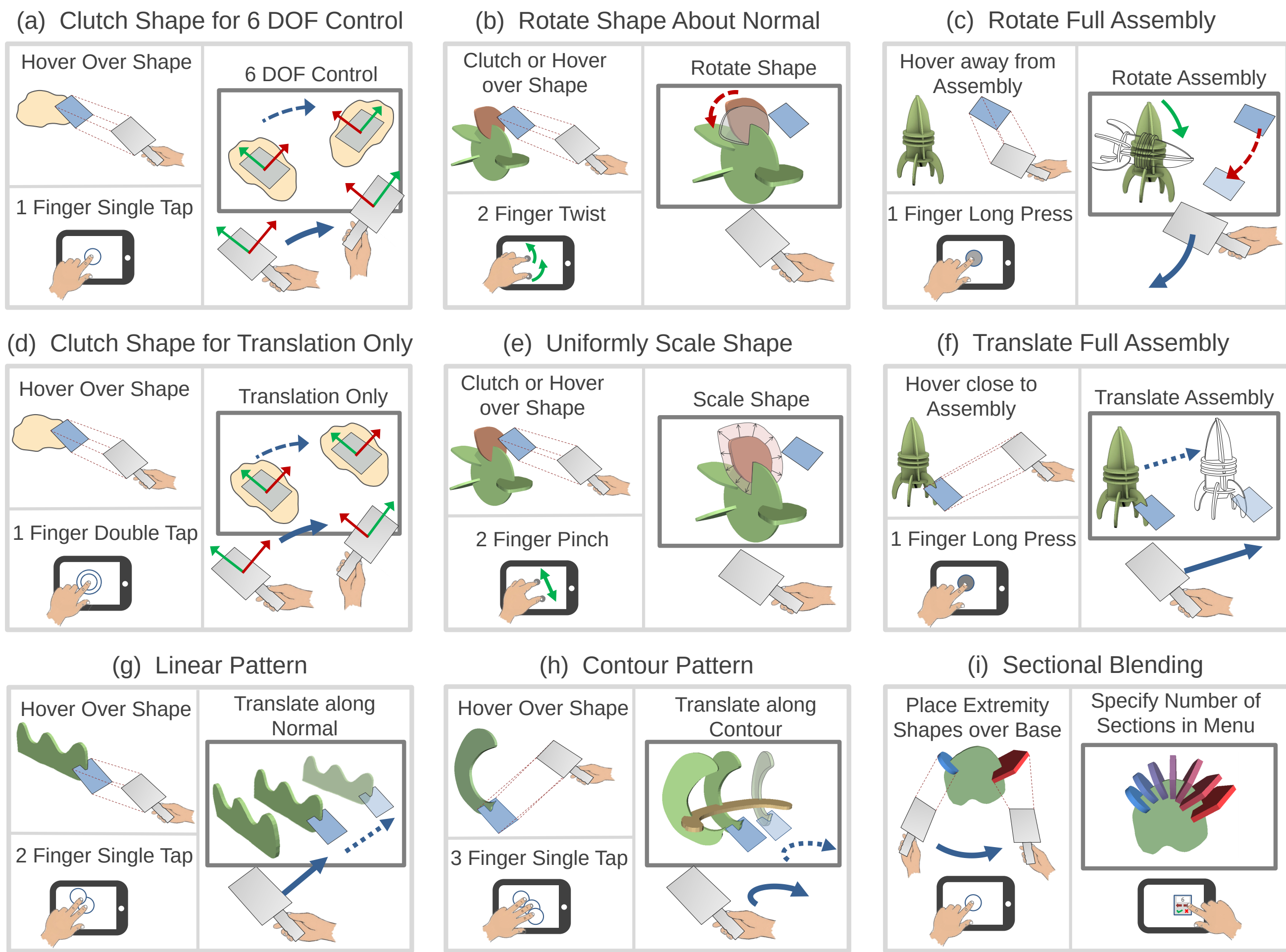

Fig. 14. 3D assembly interactions using asymmetric bimanual coordination of the proxy's motion with multi-touch gestures on the tablet. Each cell shows the hover state of the planar cursor, the multi-touch input, the proxy's motion, and the outcome in the 3D scene.

5.2.1 3D Manipulation and Configuration of Planar Shapes

Multiple copies of the shapes laid out on the desk can be individually picked up and configured into a 3D model. A clutched shape is overlaid on the cursor, such that it follows the proxy's motion (Figure 14(a)). It can then be placed at a static 3D location by simply releasing it there.

Shapes that intersect with pre-existing assembly shapes are automatically adjusted to be orthogonal to their adjoining neighbors. This ensures physical connectivity during fabrication [8] and provides a structured appearance of the model. The entire assembly can also be rotated or translated at any point for adjusting the viewing direction or gaining access to occluded regions (Figure 14(c,f)). Assembly shapes can be discarded by releasing them over the trash bin.

A clutched shape can also be manipulated along constrained DOF fine-level control. For example, if a one-finger double tap gesture (instead of single tap) is used to clutch a shape, its motion is restricted to translation only (Figure 14(d)). Similarly, by applying a two-finger twist gesture, a clutched (or asembly) shape can be rotated about its planar face normal (Figure 14(e)). This allows users to achieve difficult shape orientations without straining the wrist.

5.2.2 Procedural Operations

We enable procedural operations for geometric regularity, aesthetic design, and structural fidelity. Here, users can create a parallel pattern of identical shapes along a linear path (Figure 14(g)) or the contour of another shape (Figure 14(h)). For this, a two or three finger single tap gesture indicates the type of patterning, and copies the shape hovered by the cursor. The proxy's motion then defines the placement of the copied shape along a constrained path. Users can also create a blended pattern between two non-identical shapes (Figure 14(i)) by first placing two end point shapes over a base shape, and indicating the number of intermediate shapes in the blend menu (displayed on the tablet). The intermediary shapes are obtained by interpolating corresponding vertices between the two end-point shape profiles, similar to [8].

5.2.3 Shape Modification

By applying a two-finger pinch gesture, a clutched or assembly shape can be uniformly scaled (Figure 14(e)). Like-

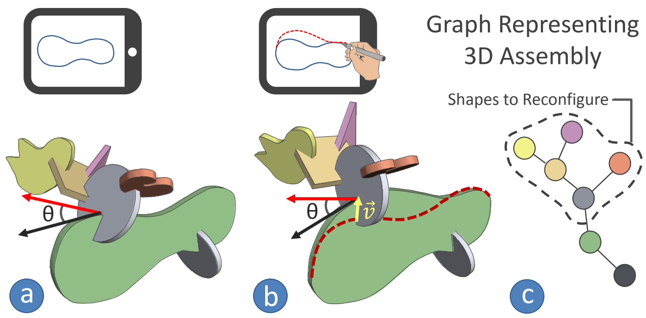

Fig. 15. (a) Initial assembly, (b) Adjusted neighbors after base shape modification, (c) Graph structure representing the assembly.

wise, when a 2D profile in the drawing interface is modified, the changes are reflected in the corresponding shapes in the assembly. This is analogous to *brushing-and-linking* in data visualization, where identical data within different spaces maintain a consistent representation [35].

When an assembly shape is modified, its neighbors are reconfigured such that it does not cut through or disconnect form the assembly. In Figure 15(a-b), when shape 1 (colored in green) is modified at a specific region, adjoining shape 2 (colored in gray) is displaced by vector $\vec{v}$, such that it maintains the same intersection point and tangential angle (θ) with shape 1. This transformation is then propagated across all shapes connecting to shape 2. To identify the connecting shapes, we apply *depth-first-search* on an undirected graph of the assembly (Figure 15(c)). Here, the nodes and edges represent the shapes and connectivity between shapes.

Shape 2 is not reconfigured if it is significantly larger ($> 100\%$) than shape 1. We impose this since larger shapes are typically used as base geometries, and displacing them could disrupt the assembly. Adjoining shapes forming a cyclic loop around shape 1 are also exempt from reconfiguration. This is to ensure that the loop structure, which could be intentional, does not get detached from shape 1.

5.3 Fabrication

As shown in Figure 16 (b-c), a 3D model constructed in *Proto-TAI++* can be physically constructed using a laser cutter and cheap material like cardboard, plywood, or acrylic sheets. To support assembly of the laser cut shapes, the system automatically inserts slit joints between adjoining shapes (Figure 16 (a)). The joint size and location are determined by first computing the intersecting region between adjoining shapes, and then removing the volume from that region. A slit joint is generally placed on the shape with larger area, unless the larger shape already has two slits in it.

6 User Evaluation and Results

6.1 User Study 1: Comparison with *FlatFitFab*

Given that *FlatFitFab* is a proven system intended for planar shape assembly, it serves as an appropriate basis for evaluating the unit operations in *Proto-TAI++*. This comparison mainly helps us assess the proxy's efficacy in configuring planar shapes in comparison to GUI based tools. It also allows us to compare the advantages of a collocated drawing and assembly space used in *FlatFitFab* with a separated multi-view system used in *Proto-TAI++*. We frame our observations in terms of drawing and assembly precision, task completion time, and user experience. Here, we recruited another group of 12 participants (9 male, 3 female) from a pool similar to Section 4. We used a within-subjects design where a training session and four tasks were performed using one of the interfaces, and subsequently repeated with the other interface. To reduce learning effects, we alternated the order in which the interfaces and tasks were presented.

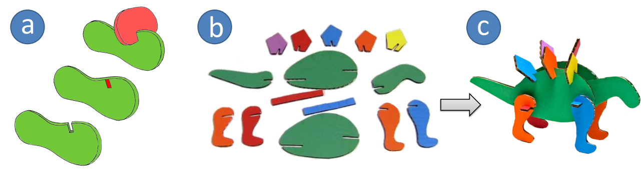

Fig. 16. (a) Automatic insertion of a slit joints, (b) laser cut shapes, (c) physical assembly built using *Proto-TAI++*.

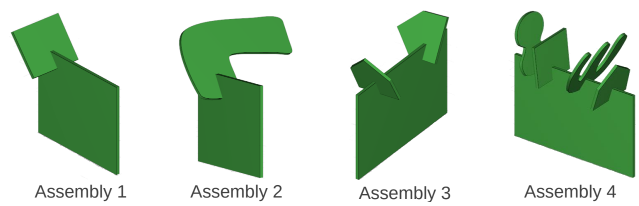

Fig. 17. Ground truth assemblies that are replicated by participants during user study 1.

Given the planar shape assembly context, the tasks involved constructing four assemblies shown in Figure 17. For each assembly, we provided a common base shape (square or rectangle) to maintain consistent physical references for measuring task accuracy. Participants were asked to draw and configure shapes mounted on this base. To cover multiple possibilities, we used 8 shapes ranging from polygons to freeform curves, and also included varying levels of assembly clutter. Participants could refer to multi-view images of the ground truth assemblies during the task. We evaluated the results based on the following factors.

6.1.1 Task Completion Time

Figure 18 shows statistical data pertaining to completion times of the four assemblies using both *Proto-TAI++* and *FlatFitFab*. We initially hypothesized that task completion for *Proto-TAI++* would take longer due to the separation of sketching and assembly modes.

Ho: The mean completion times of each assembly using the two interfaces are equal. ($\mu_{t_F} = \mu_{t_P}$)

Ha: The mean completion time using *FlatFitFab* is lower ($\mu_{t_F} < \mu_{t_P}$).

We first verified the normality of the completion time data using the Shapiro-Wilk test, and compared them for each assembly using a paired t-test. The p-values for each

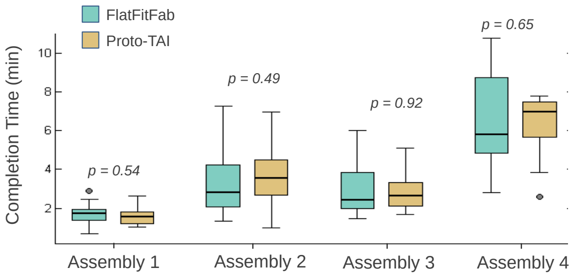

Fig. 18. Statistical distribution of task completion times using the two interfaces.

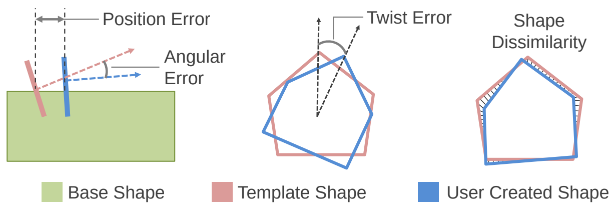

Fig. 19. Error metrics with respect to ground truth shapes: (left) positional and angular errors, (middle) twist error, (right) shape similarity.

comparison were below the significance level ($\alpha = 0.05$), indicating that we cannot safely reject **Ho**. This suggests that, despite the separation of modalities in *Proto-TAI++*, users could still draw and configure planar shapes at a rate competitive with a collocated drawing-assembly medium.

6.1.2 Task Accuracy

To quantify task accuracy, we utilized four error metrics (Figure 19) measured with respect to the ground truth shapes (Figure 17). The first three are positional, angular, and twist errors, and they collectively represent the spatial disparity between a user defined shape and the corresponding ground truth shape. Fixed references on the base (corner point, profile tangent) and ground truth shapes (profile normal and major axis) were used for measuring these errors. Similarly, the fourth error defines geometric dissimilarity between the two shapes, and is measured using Procruste's analysis [36]. It indicates how accurately users could reproduce a shape geometry using the two interfaces.

Figure 20 shows statistical distribution of the errors observed when drawing and assembling the 8 shapes using both interfaces. Here, each shape is labeled based on which assembly they lie in and their identity in that assembly (e.g. A4S3 implies third shape in assembly 4). We initially hypothesized that given the controllability of GUI interactions in *FlatFitFab*, it would result in lower error values.

H0: The mean errors using the two interfaces are equal. ($\mu_{e_F} = \mu_{e_P}$)

H1: The mean errors using *FlatFitFab* is lower ($\mu_{e_F} < \mu_{e_P}$).

Using the Shapiro-Wilk test, we checked the normality of each pair-wise difference distributions in the error data. A paired t-test was used for normal distributions, and a Wilcoxon signed-ranked test for those without. The resulting p-values are shown above their respective box plots in Figure 20. Given that most p-values for shape dissimilarity are greater than $\alpha = 0.05$, we cannot conclude that a collocated modeling space in *FlatFitFab* enabled more accurate reproduction of planar shapes.

For spatial configurations, we found that positional errors were comparable between the two interfaces. However, angular and twist errors for several shapes were larger in *Proto-TAI++*. While participants could position the shapes at 3D targets with reasonable accuracy, it was more difficult for them to define shape orientations. This could be attributed to the inability of mid-air interactions to support fine level manipulations required for orientating shapes. Mechanical constraints of the hand could also play a role in limiting the rotational movement.

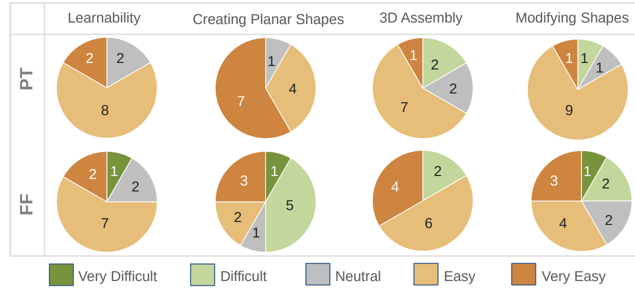

Fig. 21. 5 point Likert scale based feedback on the learnability and ease-of-use of two interfaces.

6.1.3 Subjective User Feedback

Figure 21 compares the ease-of-use of *Proto-TAI++* against *FlatFitFab*, based on a 5-point Likert scale questionnaire. While participants were able to easily learn both systems, we found notable differences in how they perceived specific components. For example, they expressed difficulty with in-situ shape creation in *FlatFitFab* due to visual clutter: *"previously drawn shapes would be in the way of seeing the current shape."* Some also pointed out that the collocated 2D-3D modeling spaces *"required me to carefully plan ahead in terms of (concurrent) placement and shape creation."* In contrast, most users found it intuitive to first draw the planar shapes on the 2D canvas and assemble them later. As one indicated, *"it was natural to follow the procedure"*. Additionally, adjusting relative proportions between shapes was found to be easier on the 2D canvas than the 3D assembly (*"it allows me to see the size differences between the things I drew and the ones I want to draw next"*). However, several participants appreciated not having to switch between sketching and assembly modes in *FlatFitFab*.

Most participants could easily assemble planar shapes in *Proto-TAI*, and appreciated the ability to transfer suggestive actions in a perceptually consistent manner: *"(by) holding the cardboard (it) was easy to control the shapes"*; *"it felt better moving the shapes in space with my hands"*; *"I could get a one-one mapping using the cardboard."* However, some also commented on the lack of precise control

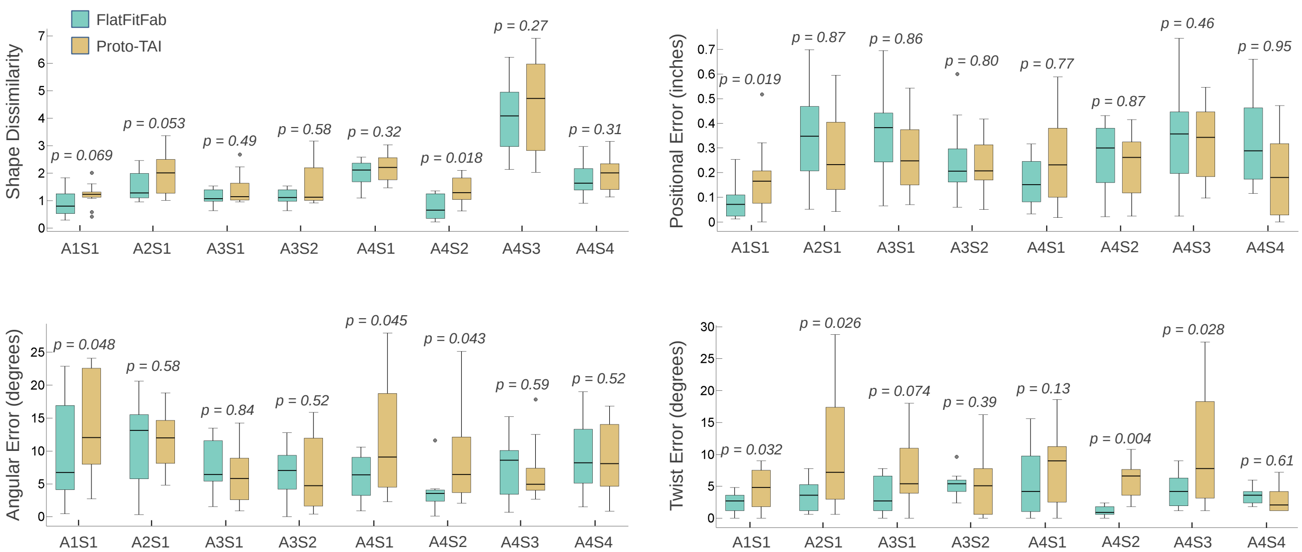

Fig. 20. Statistical distribution for errors on user defined shapes measured with respect to corresponding ground truth shapes. Each pair of box-plots shows results for a given shape using the two interfaces. Pairwise t-test results (p-values) for comparing errors are show above each pair. The shapes are labeled according to the assembly they lie on and their id within the assembly (e.g. A4S3).

with the proxy: *"I would have also liked an alternate more precise control mechanism in a few instances."* In contrast, many liked that the 3D widgets in *FlatFitFab* enabled *"precise adjustments or fine tuning of the (shape's) position."*.

6.2 User Study 2: Creative Expression

This study evaluated the utility of *Proto-TAI*++ towards creative expression of ideas. We recruited 12 (9 male, 3 female) designers from engineering and industrial design. None of them had prior experience with mid-air interactions, but most were familiar with 3D modeling. In each session, participants were given a 15 minute tutorial and familiarized with 3D planar shape assembly. They were then assigned two tasks (15-20 minutes each). The first task involved gaining familiarity with the system by constructing a familiar object like a chair, while the second task was open ended requiring them to both conceive and construct a design.

Figure 22 shows examples of models created in an average time of 10 minutes and 36 seconds (drawing: 2 min, 30 sec; assembly: 8 min, 6 sec). Here, each model is identifiable and shows more complexity and details than those created with the nominal interface [1]. This was achieved mainly by using the advanced operations added in *Proto-TAI*++.

Figure 23(a) indicates that users found *Proto-TAI*++ to support their creative expressiveness. Most were able to construct models that reasonably matched their mental image of an idea, and appreciated the ability to freely switch between drawing and assembly modes while exploring different forms. As two of them expressed: *"I enjoyed creating a concept with no formal or constraining rules"*, and *"I found myself alternating between sketching and assembly in different frequencies."* In Figure 23(b) we show that the participants were able to easily utilize the unique system features in *Proto-TAI*++.

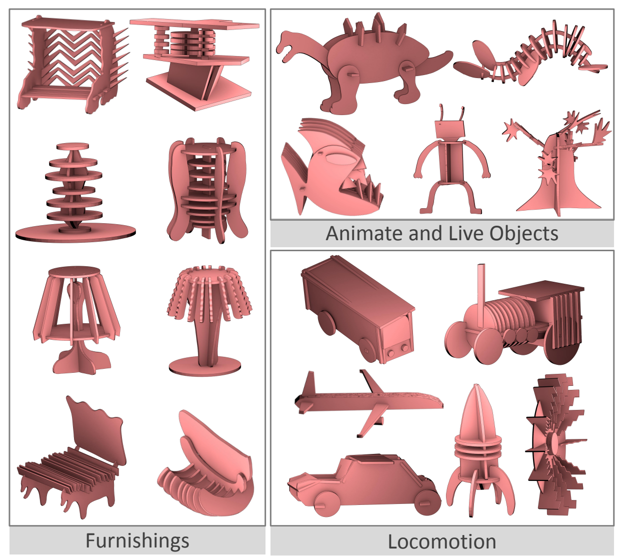

Fig. 22. Planar shape assembly models constructed by participants using *Proto-TAI*++. The models are grouped by broad level categories observed in the study.

7 Discussions, Implications, and Future Directions

Using planar shape assembly, we demonstrated how a tangible proxy can be used as a physical embodiment of virtual modeling elements in a 3D design scenario. Given the results of our studies, we find a scope for further exploring this concept using proxies with other kinds of shapes (e.g. spherical, cubical, conical etc.), material properties (e.g. density, surface texture etc.), functionalities, and semantic identities. The broader goal here is to investigate different affordances and perceptually consistent metaphors the proxies can provide within diverse 3D contexts like design, architecture, data exploration, and education.

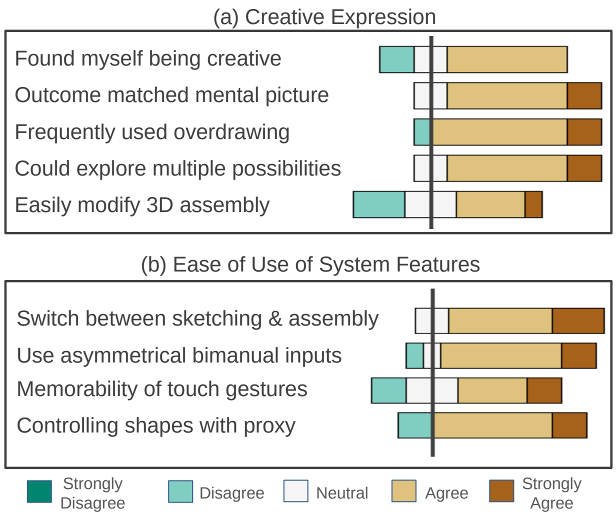

Fig. 23. 5 point Likert scale based feedback: (a) Factors influencing participants' self evaluated creative expression when using *Proto-TAI*++, (b) Ease of use of specific system features and interactions.

Given that users could easily coordinate asymmetric bimanual gestures on separate modalities, augmenting mid-air inputs with multi-touch gestures can be particularly useful when using non-instrumented proxies (or controllers) in 3D interactions. However, a more elaborate study is necessary to understand the general applicability and user perceptions on such input mechanisms. This study can include unique combinations of multi-touch media and physical proxies, with the aim to establish design guidelines, and explore interesting 3D interactive contexts that could benefit from such inputs.

While the proxy was found to provide reasonable 3D control, its precision cannot be compared to GUI tools. However, we also showed that by using context-specific constraints, freeform mid-air inputs can be interpreted as more structured 3D interactions. For example, orthogonality between adjoining shapes helped compensate for limited orientation accuracy, and enabled construction of not only identifiable but also aesthetic and fabricable models. Similarly, constrained procedural operations allowed enabled structured geometry creation using freeform motion. The utility of such constraints is apparent in the improved quality of the resulting models (Figure 22), compared to those produced with the earlier version [1]. It should be noted that this was achieved without altering our basic interactive scheme. Thus, we find value in exploring context-specific constraints within other mid-air interaction scenarios.

In 3D modeling, datum planes frequently serve as spatial references for 2D sections that define the geometry of complex 3D forms. Given the proxy's ability to arrange planar shapes in 3D space, we see potential in exploring its use as a physical datum plane that allows users to quickly arrange prominent sections of 3D surface and volumetric geometries. This metaphor extends the capabilities of *Proto-TAI*++ beyond planar shape abstractions and enable users to construct extrusions, swept surfaces, and inflation models.

Given its planarity, it would be interesting to explore the use of the tablet (or a smaller smartphone) itself as the proxy. This allows users to draw and view planar shapes directly on the proxy, resulting in a stronger correspondence between the proxy and virtual shapes. Here, the touch sensitive surface precludes the need for asymmetric bimanual inputs, and focuses all interactions on one modality. Additionally, we can also leverage the GUI display to integrate precise input via menu navigation, alphanumerical entry, sliders, and buttons.

We found that some users experienced arm fatigue after prolonged use of the proxy. This however is an inherent limitation of mid-air interactions and a trade-off for enabling direct 3D manipulation. Few users also experienced difficulty with depth perception on the flat screen display. We believe that this could be resolved using an AR based display that collocates interaction and modeling spaces.

8 Conclusions

We explored physical proxies as a means for achieving tangible 3D interactions where there is perceptual correspondence between users' input actions on the proxy and the intended outcome in the virtual 3D space. Here, we took an application based approach by demonstrating *Proto-TAI*++, a multi-modal pen-based drawing and planar proxy based mid-air assembly system for constructing planar shape assemblies in virtual space (extended version of our prior work [1]). Through user studies we showed the proxy's efficacy towards 3D manipulation of virtual shapes, and demonstrated *Proto-TAI*++ as a tool for novice designers to express creative ideas. We find this work to have broad implications in both 3D design and human computer interactions. While this paper provides a basis for extending planar shape abstractions into more complex 3D modeling contexts, the concept of perceptually consistent tangible interactions opens up interesting possibilities for novel interaction and system designs. We hope our work will lead towards a larger framework, aimed at integrating physical reality with intuitive and seamless interactions with the virtual world.

Acknowledgements

References

[1] Piya, C., and Ramani, K., 2014. "Proto-tai: Quick design prototyping using tangible assisted interfaces". In Proc. of IDETC/CIE, ASME.
[2] Fitzmaurice, G. W., Ishii, H., and Buxton, W. A., 1995. "Bricks: laying the foundations for graspable user interfaces". In Proc. of SIGCHI, pp. 442–449.
[3] Ishii, H., and Ullmer, B., 1997. "Tangible bits: towards seamless interfaces between people, bits and atoms". In Proc. of SIGCHI, ACM, pp. 234–241.
[4] Hinckley, K., Pausch, R., Goble, J. C., and Kassell, N. F., 1994. "Passive real-world interface props for neurosurgical visualization". In Proc. of SIGCHI, ACM, pp. 452–458.
[5] Arisandi, R., Otsuki, M., Kimura, A., Shibata, F., and Tamura, H., 2014. "Virtual handcrafting: Building vir-

tual wood models using tooldevice". *Proc. of the IEEE,* **102**(2), pp. 185–195.

[6] Oh, Y., JOHNSON, G., GROSS, M., and Do, E. Y.-L., 2006. "The designosaur and the furniture factory". In *Design Computing and Cognition06*. Springer, pp. 123–140.

[7] Johnson, G., Gross, M., Do, E. Y.-L., and Hong, J., 2012. "Sketch it, make it: sketching precise drawings for laser cutting". In Proc. of SIGCHI Extended Abstracts, ACM, pp. 1079–1082.

[8] McCrae, J., Umetani, N., and Singh, K., 2014. "Flatfitfab: interactive modeling with planar sections". In Proc. of UIST, ACM, pp. 13–22.

[9] Rivers, A., Durand, F., and Igarashi, T., 2010. "3d modeling with silhouettes". In Proc. of Siggraph, ACM.

[10] Tsang, S., Balakrishnan, R., Singh, K., and Ranjan, A., 2004. "A suggestive interface for image guided 3d sketching". In Proc. of SIGCHI, ACM, pp. 591–598.

[11] McCrae, J., Singh, K., and Mitra, N. J., 2011. "Slices: a shape-proxy based on planar sections.". *ACM Trans. of Graph.,* **30**(6), p. 168.

[12] Hildebrand, K., Bickel, B., and Alexa, M., 2012. "crdbrd: Shape fabrication by sliding planar slices". In Computer Graphics Forum, Vol. 31, Wiley Online Library, pp. 583–592.

[13] Willis, K. D., Lin, J., Mitani, J., and Igarashi, T., 2010. "Spatial sketch: bridging between movement & fabrication". In Proc. of TEI, ACM, pp. 5–12.

[14] Mueller, S., Lopes, P., and Baudisch, P., 2012. "Interactive construction: interactive fabrication of functional mechanical devices". In Proc. of UIST, ACM, pp. 599–606.

[15] Saul, G., Lau, M., Mitani, J., and Igarashi, T., 2011. "Sketchchair: an all-in-one chair design system for end users". In Proc. of TEI, ACM, pp. 73–80.

[16] Marquardt, N., Jota, R., Greenberg, S., and Jorge, J. A., 2011. "The continuous interaction space: interaction techniques unifying touch and gesture on and above a digital surface". In *Human-Computer Interaction–INTERACT 2011*. Springer, pp. 461–476.

[17] Huppmann, D., Luderschmidt, J., Haubner, N., Lehmann, S., Dörner, R., and Schwanecke, U., 2012. "Exploring and evaluating the combined multi-touch and in-the-air tabletop interaction space.". In VR/AR, pp. 37–48.

[18] De Araújo, B. R., Casiez, G., Jorge, J. A., and Hachet, M., 2013. "Mockup builder: 3d modeling on and above the surface". *Computers & Graphics,* **37**(3), pp. 165–178.

[19] Guiard, Y., 1987. "Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model". *Journal of motor behavior,* **19**(4), pp. 486–517.

[20] Wang, R., Paris, S., and Popović, J., 2011. "6d hands: markerless hand-tracking for computer aided design". In Proc. of UIST, ACM, pp. 549–558.

[21] Vinayak, Murugappan, S., Liu, H., and Ramani, K., 2013. "Shape-it-up: Hand gesture based creative expression of 3d shapes using intelligent generalized cylinders". *Computer-Aided Design,* **45**(2), pp. 277–287.

[22] Oikonomidis, I., Kyriazis, N., and Argyros, A. A., 2011. "Efficient model-based 3d tracking of hand articulations using kinect.". In BMVC, Vol. 1, p. 3.

[23] Song, P., Goh, W., Hutama, W., Fu, C., and Liu, X., 2012. "A handle bar metaphor for virtual object manipulation with mid-air interaction". In Proc. of CHI, ACM, pp. 1297–1306.

[24] Jerald, J., Mlyniec, P., Yoganandan, A., Rubin, A., Paullus, D., and Solotko, S., 2013. "Makevr: A 3d world-building interface". In Proc. of 3DUI, IEEE, pp. 197–198.

[25] Schkolne, S., Pruett, M., and Schröder, P., 2001. "Surface drawing: creating organic 3d shapes with the hand and tangible tools". In Proc. of SIGCHI, ACM, pp. 261–268.

[26] Mine, M., Yoganandan, A., and Coffey, D., 2014. "Making vr work: building a real-world immersive modeling application in the virtual world". In Proc. of SUI, ACM, pp. 80–89.

[27] Sheng, J., Balakrishnan, R., and Singh, K., 2006. "An interface for virtual 3d sculpting via physical proxy.". In GRAPHITE, Vol. 6, pp. 213–220.

[28] Kato, H., Billinghurst, M., Poupyrev, I., Imamoto, K., and Tachibana, K., 2000. "Virtual object manipulation on a table-top ar environment". In Proc. of Augmented Reality (ISAR), IEEE, pp. 111–119.

[29] Zhai, S., Milgram, P., and Buxton, W., 1996. "The influence of muscle groups on performance of multiple degree-of-freedom input". In Proc. of SIGCHI, ACM, pp. 308–315.

[30] Buchta, C., Kober, M., Feinerer, I., and Hornik, K., 2012. "Spherical k-means clustering". *Journal of Statistical Software,* **50**(10), pp. 1–22.

[31] LaViola, J. J., 2003. "Double exponential smoothing: an alternative to kalman filter-based predictive tracking". In Proc. of the workshop on Virtual environments, ACM, pp. 199–206.

[32] Schwarz, J., Marais, C. C., Leyvand, T., Hudson, S. E., and Mankoff, J., 2014. "Combining body pose, gaze, and gesture to determine intention to interact in vision-based interfaces". In Proc. of SIGCHI, ACM, pp. 3443–3452.

[33] Thiel, Y., Singh, K., and Balakrishnan, R., 2011. "Elasticurves: Exploiting stroke dynamics and inertia for the real-time neatening of sketched 2d curves". In Proc. of UIST, ACM, pp. 383–392.

[34] Baudel, T., 1994. "A mark-based interaction paradigm for free-hand drawing". In Proc. of UIST, ACM, pp. 185–192.

[35] Eick, S. G., and Wills, G. J., 1995. "High interaction graphics". *European journal of operational research,* **81**(3), pp. 445–459.

[36] Gower, J. C., 1975. "Generalized procrustes analysis". *Psychometrika,* **40**(1), pp. 33–51.