

SymbiosisSketch: Combining 2D & 3D Sketching for Designing Detailed 3D Objects in Situ

Rahul Arora¹ Rubaiat Habib Kazi² Tovi Grossman² George Fitzmaurice² Karan Singh¹

¹University of Toronto
Toronto, Canada

{arorar,karan}@dgp.toronto.edu

²Autodesk Research
Toronto, Canada

{firstname.lastname}@autodesk.com

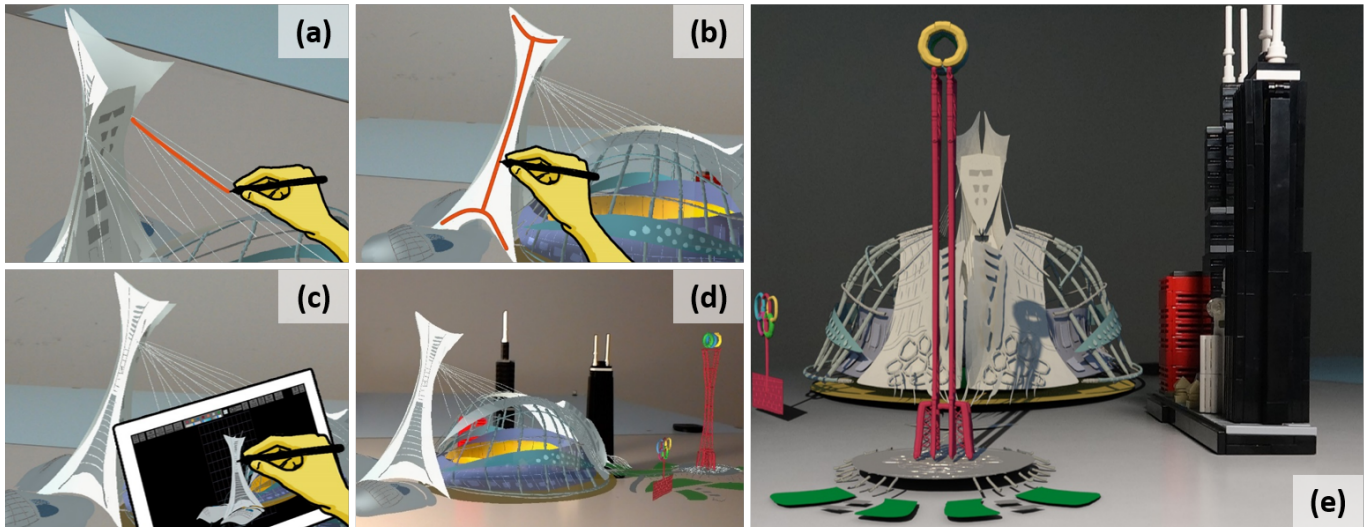


Figure 1. SymbiosisSketch combines mid-air 3D interactions (a) with constrained sketching. Users can create planar or curved *canvases* (b), and use a tablet (c) to sketch onto them. Designs are created in situ, in context of physical objects in the scene (d), allowing quick post-processing operations to seamlessly blend the virtual objects into the real world (e).

ABSTRACT

We present SymbiosisSketch, a hybrid sketching system that combines drawing in air (3D) and on a drawing surface (2D) to create detailed 3D designs of arbitrary scale in an augmented reality (AR) setting. SymbiosisSketch leverages the complementary affordances of 3D (immersive, unconstrained, life-sized) and 2D (precise, constrained, ergonomic) interactions for in situ 3D conceptual design. A defining aspect of our system is the ongoing creation of surfaces from unorganized collections of 3D curves. These surfaces serve a dual purpose: as 3D *canvases* to map strokes drawn on a 2D tablet, and as shape proxies to occlude the physical environment and hidden curves in a 3D sketch. SymbiosisSketch users draw interchangeably on a 2D tablet or in 3D within an ergonomically comfortable canonical volume, mapped to arbitrary scale in

AR. Our evaluation study shows this hybrid technique to be easy to use in situ and effective in transcending the creative potential of either traditional sketching or drawing in air.

ACM Classification Keywords

H.5.1. Information Interfaces and Presentation (e.g. HCI): Artificial, augmented, and virtual realities; I.3.6 Computer Graphics: Interaction techniques

Author Keywords

3D drawing, design sketching, augmented reality

INTRODUCTION

Sketching has been a design and conceptualization aid for centuries. While traditional 2D sketching is ingrained into our consciousness, abilities, and training from childhood [56], the growth of consumer grade augmented and virtual reality (AR/VR) devices has recently enabled us to transcend the limits of the sketch canvas digitally—equipping artists with the unprecedented ability to sketch 3D curves directly in the air [22, 35, 51]. 3D sketching is indeed becoming an increasingly popular medium of art and storytelling [20, 4, 1].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CHI 2018, April 21–26, 2018, Montréal, QC, Canada.

Copyright © 2017 ACM ISBN 978-1-4503-5620-6/18/04 ...\$15.00.

<http://dx.doi.org/10.1145/3173574.3173759>

Despite their novelty, current mid-air 3D drawing systems lack the precision and constraints necessary for use in conceptual design. This is primarily due to our ergonomic inability to draw well and in detail at arbitrary 3D scales, orientations, and without a physical surface to support and steady our sketching motion [7, 34]. Interestingly, traditional and digital 2D sketching systems address this issue well. They are ergonomically superior in comfort, not fatiguing to use, and afford greater control and precision over sketch strokes. Over the centuries, 2D sketching has further developed a vocabulary (see Figure 4) of illustration styles, textures, and rendering techniques to depict geometric detail, material, and lighting [26, 42]. While drawing such illustrative detail in 2D is as common and efficient as handwriting, executing regular stroke patterns mid-air in 3D is extremely challenging (see Figure 2).

The biggest problem with 2D sketching is conveying the absolute 3D depth from a single viewpoint using only a flat surface. Often multiple 2D sketches from disparate viewpoints, using construction lines [55], perspective grids [9] and scaffolds [46], are required in 2D to convey all visual aspects of a 3D design. In contrast, depth is trivially integral to mid-air 3D sketching.

Another common practice in 2D design sketching is the use of photographs or renderings of 3D objects as a reference to situate sketches in a 3D context [21]. Augmented Reality has the potential of incorporating a 3D context directly into the design process. However, AR exacerbates the problem of visual clutter when complex designs are viewed as a collection of 3D curves. Creating surfaces from 3D curves is typically treated as a separate and subsequent stage in conceptual design [11]. Integrating surfacing with curve sketching can provide occlusion primitives to reduce visual clutter from both the real environment and sketched 3D curves, as well as a local 3D surface to map 2D sketch strokes.

We distill these synergistic insights into SymbiosisSketch, a system that judiciously combines 3D mid-air and 2D surface sketching to produce conceptual models comprising 3D curves and surfaces within an AR setting. We equip the designer with a HoloLens, a spatially tracked 3D stylus, and a digital tablet to view and sketch both mid-air in 3D and on a 2D surface (see Figure 6). The user specifies design surfaces by sketching multiple strokes. Working in AR allows designers to work directly with existing 3D objects in situ. Finally, our system allows users to rescale work volumes, allowing drawings that range from miniature to larger than life in size.

Our principal contribution is a novel hybrid sketching workflow in AR with tools and widgets that leverage the complementary strengths of 3D mid-air and 2D tablet sketching. We integrate surface creation early in our drawing workflow, as *drawing canvases* for 2D sketching, and better visualization of the evolving design.

We performed a user study with both professional designers and experienced amateurs to evaluate our system. Our evaluation and results confirm the hypothesized advantages of our hybrid workflow for 3D conceptual design.

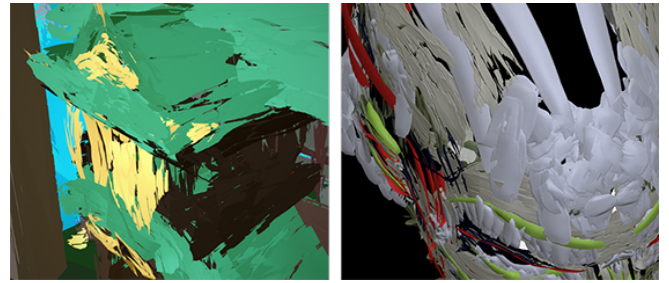


Figure 2. Lacking surfaces, Tilt Brush users employ many strokes to depict texture or the illusion of a surface. ©Kamikakushi de Sen et Chihiro (left) and Christopher Watson (right). Used under CC-Attribution 3.0.

RELATED WORK

Our work builds on a body of existing research in freeform and surface-constrained sketching techniques, tablet-aided AR/VR tools, and digital design in context.

Freeform 3D Design Tools and Studies

One of the earliest examples of freeform 3D modelling is HoloSketch [22], that supported 3D creation of primitives and freeform tube and wire geometries. Despite its novel interface, the system evaluation reported issues with a lack of motor control and the resulting noisy strokes plaguing mid-air 3D drawing, further corroborated by recent studies [7, 34] conducted with contemporary hardware. Inspired by HoloSketch, CavePainting [35] utilizes a cave environment, and Tilt Brush [4] a VR environment for 3D sketching. Despite the focus on creating 3D strokes, users naturally use multiple nearby strokes to depict surfaces. While a “guides” tool in Tilt Brush allows the projection of 3D strokes onto a pre-defined set of surfaces, executing accurate 3D strokes remains challenging [7]. Furthermore, users are restricted to the predefined surface shapes, and cannot create ones of their choice. In contrast, we empower users to author 3D surfaces and detail them using a mix of 2D and 3D sketching. This makes our tool more suitable for design sketching.

Systems aimed at the creation of CAD models in VR employ various curve and surface fitting techniques for precise 3D models [27, 53]. Gravity Sketch [1] is a recent commercial tool with bimanual interactions to provide a CAD-like workflow in VR. While these systems target precise CAD models, our focus is early concept design, epitomized by loosely constrained sketches with rich illustrative detail. SymbiosisSketch surfaces lie in-between surface patch networks used in CAD, and very loose hand-swept surfaces created by existing VR/AR drawing tools such as Surface Drawing [45].

Surface-constrained Sketching Techniques

Digital 2D sketching is more than half a century old [49]. Since 2D strokes are inherently ambiguous for describing a 3D curve, such systems employ UI tools, or use geometric principles to infer the 3D surfaces to project input 2D strokes on. Common techniques include using coordinate planes or extruding profile curves in principal directions [8], using two curves sketched in orthogonal views to define a surface [28], utilizing a custom-designed physical prop for posing sketch surfaces [43], or minimizing an energy functional targeting

specific design scenarios [32, 37]. These techniques assume relatively structured and accurate 2D stroke input, where the desired 3D surface is either explicit by construction or can be inferred by surface fitting. Parametric surface fitting to 3D curves is a well studied problem in computer graphics [5, 44], typically focused on input 3D curves that form the exact boundaries or principal curvature lines of smooth surface patches. In contrast, we present a 3D surface fitting formulation suited to imprecise, unstructured and noisy collections of mid-air 3D strokes [7], where input curves may not intersect as expected, or may contain other inconsistent surface cues.

Utilizing a Tablet in Immersive AR/VR

Early research in VR interaction compensated for a lack of high fidelity 3D tracking by using 2D interfaces to interact with UI elements rendered in VR. Angus and Sowizral [6] explored the use of a virtual 2D tablet for quicker interaction and positioning in VR applications. Bowman et al. [13] targeted the manipulation of VR terrains, a task naturally suited to 2D interaction. The virtual rendering of a physical tablet has been used to present menus in VR [14], for object manipulation [50], and for 3D selection and manipulation [40]. A virtually rendered physical tablet in VR however, cannot presently replicate the latency-free high visual fidelity needed for 2D sketching. Therefore, we use a physical tablet, motivated by Dorta et al.’s [24] use of a tracked tablet to draw on virtual planes. However, our approach overcomes the reported ergonomic difficulty in accurately manipulating a tablet in 3D, and also allows drawing accurate non-planar strokes. In addition, we designed our system in AR to allow drawing and visualization in context, on a physical tablet, and on 3D objects.

Designing in Context

Objects are designed to serve a function, often interacting with people and objects in an environment. A number of 3D sketching systems [16, 15, 21, 31, 33, 36, 39, 41, 57] thus leverage existing visual media, 3D models, or spatial information of the environment to inspire and guide the design.

The success of a 3D sketching system using sketch planes depends to a large extent on the speed and accuracy with which users can choose desired sketch planes [9, 33]. A physical tablet has been shown to be effective in mapping directly to a subset of physical sketch planes in 3D [38]. We present an indirect mapping of the tablet in AR to capture physical planes in arbitrary position, orientation, and scale, allowing users to draw comfortably in situ.

While the use of AR further allows our designers to draw in 3D on physical objects, our surfacing tool greatly enhances this capability by allowing 3D scribbles on physical surfaces to turn them into *drawing canvases* for 2D sketching.

MOTIVATION: HYBRID INTERFACES FOR DESIGN

SymbiosisSketch is motivated by ideas from multiple and complementary disciplines—the wealth of 2D drawing practice, modern VR-based design tools, in situ 3D design research tools, and the affordances of modern AR hardware.

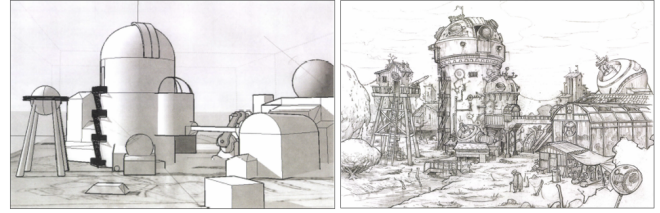


Figure 3. Design workflow using 3D and 2D tools. Rustam Hasanov modeled an underlay of an interior scene in a 3D modeling tool (SketchUp), and then drew details over the top of those underlays using a 2D interface and tools. ©Rustam Hasanov. Used with permission.



Figure 4. Details in 2D drawings: structural details, architectural embellishments, and textures. © Jake Parker (left) and Johannes Figlhuber (right). Used with permission.

Portraying 3D shapes using 2D

3D Tools for Layout and Perspective

While 2D sketching is a ubiquitous medium for drawing and conceptualization, depicting 3D shapes in 2D requires expertise and experience. In particular, understanding, controlling, and portraying depth in 2D is difficult [47]. Experienced designers often employ the complementary strengths of 2D and 3D design tools in their workflows [48] for detailed drawings. For instance, 3D modeling programs can aid in the basic layout of perspective drawing grids, quickly blocking out proportion and composition, from a desired viewpoint, before exploring design details in 2D (see Figure 3).

2D for Surface-constrained Strokes, Texturing, and Details

As evident in numerous books on drawing and design [26, 42], 2D sketching incorporates a myriad of artistic styles and textures, that are used to convey structure, material properties, function, lighting, and even motion (see Figure 4). While commonplace in 2D, authoring such detail using freeform 3D curves without a constraining physical surface is tedious, tiring, and error-prone, further motivating the use of a physical tablet.

Designing in Situ

A central idea of our work is the use of AR to visualize and interact with a physical 3D stylus, a tablet, and objects in context with the utmost fidelity. Designing in context of the physical world also requires working with scales spanning from the miniature to the massive. To this end, our tool should allow designers to interact minimally with real world shapes and contours, minimize visual clutter, draw with a comfortable posture, and still visualize their creations 1:1 in the real world.



Figure 5. Post-processed results. HoloLens hardware limitations restrict us to basic Gouraud shading. With appropriate shading and occlusion, SymbiosisSketch designs can seamlessly blend into the real world. (Clockwise from top) war helicopter shooting at Captain America (Figure 14a), mechanical wing augmentation (Figure 14c), mini car (participant creation), Flintstones’ house (author creation), and large fan (Figure 14d).

SYSTEM OVERVIEW AND SETUP

Our system requires three main hardware components: an AR-capable Head-Mounted Display (HMD), a tablet for 2D drawing, and a digital pen with 6-degree of freedom (DoF) motion tracking to draw mid-air, as well as on the tablet. The pen position and orientation is tracked using Vicon motion capture cameras. The three devices communicate in real-time to ensure a unified, consistent, state of the world and the design elements. We used the Microsoft HoloLens as our AR HMD, whose spatial mapping capability gives us a coarse 3D map of the physical world as a triangle mesh. The tablet for 2D interface is a Microsoft Surface Book clipboard, detached from its keyboard.

Metaphorically, our setup is analogous to a painter’s bimanual interaction with her tools. Typically, a painter reserves her dominant hand for holding the paintbrush for making marks on the canvas, and the non-dominant hand to hold the color palette. In our multimodal interaction system, the user holds the pen in the dominant hand, while the tablet is typically held in the non-dominant hand (but can also rest on a table like an easel). The tablet serves two purposes. First, it renders an orthographic view of the world, and users can draw on the tablet to project their strokes to a 2D *canvas* visible from this orthographic window (Figure 6a). Second, it also includes a function toolbar, accessible with pen or touch input (Figure 8). For a given *drawing canvas* planar projected on the

tablet screen, the position of the orthographic camera remains fixed for comfortable sketching, regardless of the position and orientation of the tablet. Hence, we do need to not explicitly track the position and orientation of the tablet.

In our prototype, we use the buttons on a standard mouse to trigger these interactions. The mouse is magnetically fastened to the back of the tablet, allowing the user to interact with the mouse while holding the tablet (see Figure 6b). Alternatively, the user can also hold the mouse by itself in the non-dominant hand, while the tablet rests on a table.

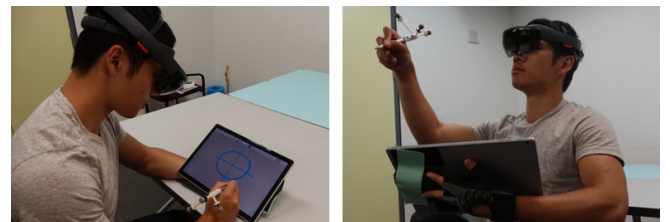


Figure 6. Setup: the user puts on the HoloLens and draws with a motion-tracked stylus, on a tablet (left), or mid-air (right) using a mouse affixed to the back of the tablet.

COMPONENTS

We first introduce some terminology, and the basic interface and interaction components of our 3D design tool.

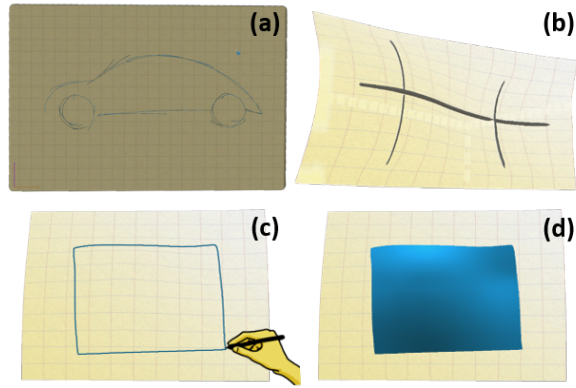


Figure 7. Strokes drawn using the 2D tablet are projected onto *drawing canvases*. (a) A *planar drawing canvas* is a rectangle with the same aspect ratio as the tablet. (b) Surfaces form *curved drawing canvases*. (c) Users can draw closed curves on *canvases* to define (d) *solid surfaces* which also lend occlusion, lighting, and shadows to the design.

Strokes

In practice, 3D strokes are commonly stored and rendered either as *ribbons* (flat narrow sheets), or as *tubes* (generalized cylinders). We use tube rendering for both 2D and 3D strokes to provide a homogeneous sketch appearance, and to distinguish curves from surface patches (*drawing canvases*).

Drawing Canvases

Most 3D objects are better represented, using a set of surface patches, than just a collection of curves. In programs like Tilt Brush, artists tediously align many wide ribbon-shaped strokes next to each other to create an approximate impression of a surface (see Figure 2). Most 3D sketching tools do not address surface creation at all, and some define transient surfaces on which to draw curves and subsequently delete the surface [9]. In contrast, our output is a collection of 3D curves and surface patches. In addition to defining the 3D conceptual design, our surfaces map to the physical tablet as *drawing canvases*. The design of our *drawing canvases* is motivated by similar concepts utilized in existing creative works. For example, the unique art style of the Academy award winning short film *Paperman* [54] was accentuated by sketching onto modelled surfaces. A similar concept was also employed by Bassett et al. [10] for authoring detailed painterly characters. Technically, a *drawing canvas* in our system is either a planar rectangle, or a surface patch represented as a height field over a planar rectangle.

Solid Surfaces

A *solid surface* is an enclosed region projected to a *drawing canvas*, with a uniform material appearance. Essentially, this is a triangulated mesh that aids in 3D model understanding by providing shading, occlusion, and shadows [19, 30]. In the absence of *solid surfaces*, depth perception can be poor and complex objects appear to be “stroke spaghetti” instead of coherent 3D objects, even with binocular cues [7].

USER INTERFACE & INTERACTIONS

We designed SymbiosisSketch to enable a symbiotic relationship between 2D and 3D sketching, utilizing the strengths of

both, while minimizing their respective shortcomings. Mid-air strokes and UI manipulations are executed by moving the pen with the dominant hand, while holding the mouse buttons with non-dominant hand. The 2D tablet renders the 3D world. Any stroke drawn in 2D is mirrored in the 3D world in real-time, and vice-versa. The 2D interface (see Figure 11) shows an orthographic projection of the active *drawing canvas*, a color palette, and other functionality and configuration settings (also see Figure 8). Figure 1 demonstrates the overall workflow of our system.

Sketching in Freeform 3D

For freeform 3D strokes, the user presses and holds the left mouse button to initiate mid-air sketching. The stroke drawn traces the motion of the pen’s tip. We overlay a virtual pen-tip to aid positional perception.

Drawing Canvases

Creating and manipulating *drawing canvases* (both planar and curved) is a defining aspect of our system that allows our users to sketch with greater precision. Users can define and use an arbitrary number of *drawing canvases* during their designs, but only one *canvas* is active at a time. The 2D tablet’s viewport always projects an orthographic view of the active *drawing canvas*, allowing users to draw on the tablet and project their strokes onto that *canvas* in 3D.

Planar Drawing Canvas

Initially, the system displays a *planar drawing canvas* at a fixed location in 3D. The designer can thus sketch planar strokes without foreshortening on the *drawing canvas* by sketching on the tablet screen. We constrain the aspect ratio of the *planar drawing canvas* to match the tablet screen (Figures 6a, 7a). At any point in time, the designer can activate a *planar drawing canvas*, and position it in 3D space.

Curved Drawing Canvas

To create a *curved drawing canvas*, the user presses the “Define new canvas” button in the tablet toolbar. The user then draws a few canvas-defining 3D curves, coarsely specifying the intended surface in 3D space (Figure 9). Once completed, the user presses “Generate Canvas” to generate a curved surface that best fits the canvas-defining 3D strokes. The tablet’s orthographic camera is positioned in front of the surface, ensuring the maximal projection area on the tablet plane, to maximize *sketchability* [7, 8] (see Figures 7b, 11a). It is worth noting that these canvas-defining strokes need not always be drawn mid-air. A user can freely combine canvas-constrained strokes (drawn using 2D tablet) with freehand mid-air 3D strokes to define an intended *drawing canvas*. This is very useful for creating *canvases* for depicting details, which can be drawn efficiently over existing *canvases* (see an example in Figure 14a).

Technically, our *drawing canvas* is an open surface with a continuous boundary topologically equivalent to a disc. This class of surface patches are most common in 3D surface modeling, and further allows us to project strokes drawn on a tablet onto the entire *canvas* without frequent view manipulation.

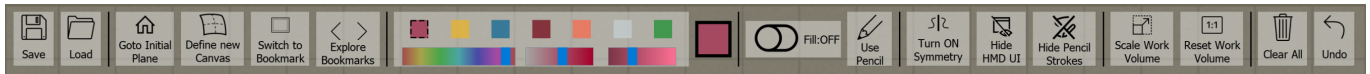


Figure 8. 2D UI toolbar. (left to right) save/load, initial *canvas*, create *canvas*, explore and select bookmarks, color tools (palette, HSV sliders, indicator), fill, pencil/ink toggle, symmetry plane, UI and pencil toggles, workspace scaling and reset, clear all and undo stroke. Buttons are shown only if currently useful. For example, selecting the pencil hides color tools. Icons © icons8.com. Used under CC BY-ND 3.0.

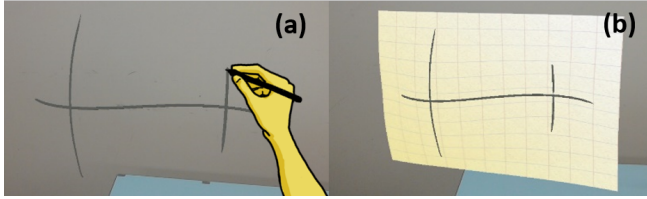


Figure 9. Creating a curved drawing *canvas*. (a) The user draws a few strokes using the 2D and/or 3D interface. (b) A surface patch is fit to these strokes.

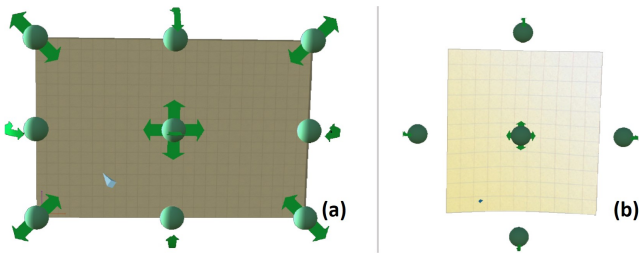


Figure 10. Widgets for direct 3D manipulation: translation and rotation (all *canvases*), and scaling (*planar canvases*): shown at the center, edges, and corners, respectively.

By default, the *canvas* is clipped against the bounding rectangle of its *canvas*-defining strokes on the best-fit plane, giving it the appearance of a rectangle curved in 3D. However, if the projection of the largest enclosing *canvas*-defining stroke is approximately a closed simple curve, we further trim the *canvas* against this stroke. While the surface behaves just like a rectangular *canvas* for drawing, we hypothesized that the trimmed surface is visually more representative as part of the 3D design process.

Canvas Manipulation

Our *canvas* manipulation tools enable designers to freely transform an active *drawing canvas* in 3D space (Figure 10). In our tool, to move a *drawing canvas*, the user positions the pen's tip inside the translation widget, then presses and holds the right mouse button with non-dominant hand, and then freely moves it in 3D space. Similarly, we provide widgets along the *canvas* edges to scale (*planar canvas* only) and rotate the

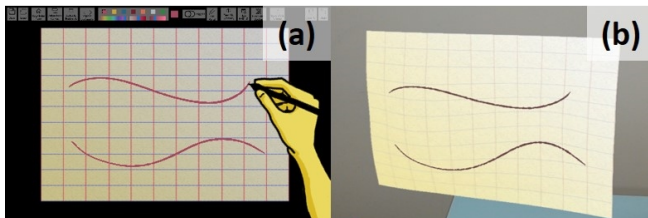


Figure 11. (a) The user sketches in the 2D view of the *canvas*. (b) The corresponding strokes in 3D.

canvas about its local origin (see Figure 10). Since accurately rotating in 3D is difficult, we snap these rotations to multiples of 15° .

Canvas Bookmarking

Our system automatically bookmarks all *drawing canvases* utilized during the design. The toolbar UI allows switching to previously utilized *canvases* for design iterations.

Sketching on the 2D Tablet

Our tools can project a broad class of curved surfaces or arbitrary planes in 3D to the tablet's 2D screen. The tablet displays an orthographic view of the active *drawing canvas* (see Figure 11a). Drawing on the tablet projects the input points onto the 3D *canvas* to create a new stroke. The ability to draw on an intended surface using the 2D physical surface discounts the depth error of mid-air 3D strokes [7], which is a major source of drawing inaccuracy.

When the pen is in proximity of the tablet's screen, we display a virtual pointer at the projected position in 3D. This indicator gives artists the ability to quickly judge the expected 3D position of their stroke when drawing in 2D. A grid texture on the *canvas* further helps users draw precisely. When the pen leaves the tablet's vicinity, the virtual pointer snaps back to the pen's tip.

Solid Surface Tool

The user can toggle ON the "Fill" button on the toolbar to activate the solid surface mode. In this mode, when the user draws a stroke on the tablet, our system automatically closes that curve (by connecting the first and last points), triangulates the region, and projects the shape onto the active drawing *canvas* with a selected solid color (see Figure 7c–d). The solid surface tool is only accessible by the 2D tablet interface, since executing surface-constrained curves mid-air is challenging and error-prone.

Workspace Scaling

AR/VR environments allow designers to draw and visualize at any scale. However, drawing at extreme scales—even if humanly possible—is detrimental to stroke quality [7]. Especially when drawing in situ, the user often needs to design at such uncomfortable scales.

In order to mitigate these issues, we provide a workspace translation-cum-scaling tool. By default, the user works in a 1:1 scale, which we call the canonical workspace. On activating the tool using the tablet toolbar, the user sees two rays—one emanating from her head and another from the pen. Pressing the right mouse button creates a 3D point (q_1), which approximates the intersection of the two rays. This is one corner of the intended workspace. Performing this action again defines the diagonally opposite corner (q_2). Our point

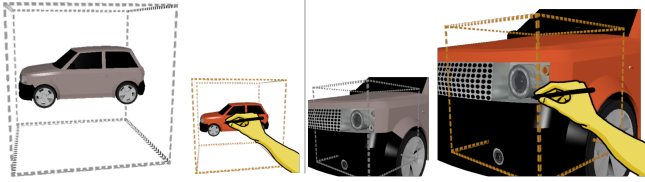


Figure 12. Workspace scaling tool for drawing a car. The user uses a large workspace to draw the shape of the car in a small, comfortable, scale (left). She then defines a small workspace centered on a headlight (right). This “zoom in” effect allows her to capture the details of the headlight’s shape, define a new *canvas* on it, and draw highlights.

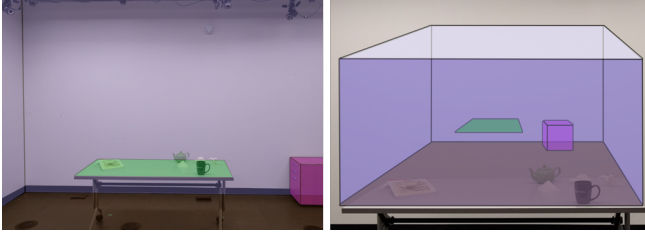


Figure 13. Physical planes detected by the HoloLens are automatically bookmarked (shown overlaid on the scene, left). With workspace scaling, users can visualize and work even with room-scale planes in an accessible position and comfortable scale (right).

selection method is inspired by Cheng and Tataksuk’s [17] virtual pointer.

We then map this physical workspace to the canonical volume, which provides a virtual view of the world (orange dashed boxes in Figure 12). The user can now draw comfortably with respect to the canonical workspace, with results mapped to the defined workspace (dashed gray boxes in Figure 12).

Interaction with Physical Objects

Being an AR-based tool, SymbiosisSketch directly allows users to interact and draw relative to 3D objects in the physical world (see Figure 14). Drawing directly over many physical objects however, can be awkward, due to their steep curvature (see Figure 15c), softness (drapery or clothes, see Figure 17), or size and reachability (buildings, ceilings, or cars, Figure 14d). In such cases, defining *curved canvases* using physical objects as proxies, or using workspace scaling, can greatly reduce the direct interaction required with the objects, improving final design outcomes.

The above capability is further bolstered by the HoloLens’s capability to map the physical world as a 3D triangle mesh, which we utilize to capture physical planes from the real world. Ideally, we would like to automatically use this mesh to generate *canvases* over all interesting physical objects. Unfortunately, the resolution of HoloLens’s spatial mapping mesh is not fine enough to capture small objects or curved surface detail. We thus only utilize the physical planes provided by the HoloLens Spatial Mapping API [2]. These physical planes are automatically bookmarked, and particularly useful for precise in situ architectural and interior design. The user can directly access physical landmarks such as the floor, ceiling, tables, and walls, and bring them into the canonical workspace for further sketching (see Figure 13).

Helper Tools

Pencil Tool

The pencil tool is used to draw strokes of a pre-defined thickness and color. It works just like the regular stroke tool, but users have a switch to hide all pencil strokes. In conjunction with *canvas* bookmarking, it serves as a simple alternative to layers. For example, a user may use the pencil to draw out scaffolds and other construction lines, and finally hide the pencil strokes in the finished design.

Symmetry Plane

The symmetry plane mirrors the users’ strokes about itself. We provide translation and rotation widgets on the symmetry plane for positioning and orientation.

Additional Sketching Tools

We also provide a few essential sketching tools to help users utilize our system efficiently. Users can manipulate the stroke width using the mouse wheel. Stroke color can be modified from a pre-defined palette, or using HSV sliders. Since the additive display of the HoloLens cannot render true black, we clip the value slider to disallow very dark colors. We also provide an undo tool. Finally, the resulting designs can be exported as an .OBJ file for post-processing.

IMPLEMENTATION DETAILS

Our system is implemented in two parts: the tablet app is written in C++ using Qt and OpenGL, and runs on a Surface Book with a 2.60GHz Intel i7 CPU and Intel 520 GPU and support for pen and touch inputs. The second application runs on the HoloLens, and is written in C# using the Unity engine. While the HoloLens is capable of tracking objects, the latency and precision are not good enough for our task. We thus augment our system with a Vicon motion capture system connected to a computer to track the drawing pen in the air. The tablet app communicates with the motion capture server at 100Hz, and all information exchange between the tablet and HMD is carried out over a reliable low-latency TCP connection. We now provide technical details for implementing various pieces of the system.

Stroke Smoothing and Reparametrization

We smooth the input strokes using a real-time smoothing procedure inspired by Elasticurves [52]. Let $\{s_i\}$ be a sequence of input points, and $\{p_i\}$ be the sequence of points forming the stroke, after real-time smoothing. Then,

$$p_i = \begin{cases} s_i & \text{if } i \in \{0, n\}, \\ p_{i-1} + f(s_i - p_{i-1}) & \text{otherwise.} \end{cases}$$

Here, n is the number of points in the sequence, and $f \in [0, 1]$ is the smoothing factor. The stroke starts and ends exactly at locations specified by the user, with smoothing in the middle modulated by f and the drawing speed. We set $f = 0.8$ for tablet strokes, and $f = 0.1$ for mid-air strokes (for less and more smoothing). We then reparametrize according to arc-length to make the strokes more suitable for *canvas* fitting and rendering. Finally, 10 iterations of bi-Laplacian smoothing remove additional jitter from the mid-air 3D strokes.

Rendering Curves and Solid Surfaces

For rendering curves as 3D tubes, we need to compute an orthogonal 3D frame at each segment of the stroke, where one of the three orthogonal directions is the tangent to the curve at that point. We start with an arbitrary frame at the first segment, and use parallel transport (Hanson and Ma [29]) to compute the Bishop frame [12] for the rest of the curve. The thickness of mid-air curves is given simply by the current stroke width setting set by the user. For tablet strokes, this value is multiplied by the normalized pressure value at each point, lending a typical *sketchy* look to the strokes.

When the “Fill” tool is active, trimmed solid surfaces are rendered by computing a Constrained Delaunay triangulation (CDT) with all the segments of the input stroke (including an one joining the end-points) as constraints. We compute the CDT using the standard technique (Cheng et al. [18]): first compute the CDT on a convex region covering the stroke, and then remove the triangles outside the region enclosed by the stroke. The vertices of the triangulation are then projected onto the active *canvas*, similar to the boundary curve points.

Curved Canvas Fitting and Drawing

The surface fitting function proceeds in two stages: finding the best-fit plane for the set of strokes used to create the surface, and then fitting a smooth height field to the strokes, relative to the best-fit plane. For the best-fit plane, we first uniformly sample a fixed number of points from each input stroke. We then solve for the best-fit plane in a least-squares sense [25]. This plane’s local +Y and -Z axes then trivially define the orthographic view’s up and forward directions.

To define the *curved canvas*, we then fit a height field (from the best-fit plane) to the input strokes by minimizing the field’s thin plate spline energy. Since the input 3D curves can have inaccuracies, we use the approximate fitting method proposed by Donato and Belongie [23]. Their method utilizes a regularization parameter λ , where $\lambda = 0$ means that the solver tries to fit exactly to the input points, while a higher value implies sacrificing exact fit for smoothness.

Since we expect curves input via the 2D interface to be very precise, we set λ to zero for those, while for the curves drawn mid-air, we set $\lambda = 0.1$. Finally, we scale the viewport of the tablet’s camera such that the whole *canvas* is visible from it.

Workspace Translation and Scaling

For selecting 3D points for defining the workspace, approximate intersection between the pen and head rays is defined as the mid-point of the shortest line segment between the two rays. Once the workspace is defined, it maps an input position \mathbf{p}_c in the canonical volume to the new volume as $\mathbf{p} = s\mathbf{p}_c + \mathbf{t}$, where $s = \|\mathbf{q}_1 - \mathbf{q}_2\|$, $\mathbf{t} = (\mathbf{q}_1 + \mathbf{q}_2)/2$.

USER EVALUATION

We evaluated our system with professional as well as amateur illustrators and designers. The goals of our study were to observe user workflow, test the system’s core functionality, and to understand its limitations. We also include a 3D mid-air drawing condition to compare purely 3D mid-air sketching with SymbiosisSketch.

Participants

We recruited six participants (age 27-46, 2 female) for our study. Participants were experienced with graphic design (*P1*, *P5*), 3D modelling (*P2*, *P6*), and VR-based design tools (*P1-2*, *P5-6*). Participants were compensated for their time.

Procedure

The study was conducted in our lab. Each evaluation session was 90-120 minutes long, and consisted of four phases.

Overview and training (10-20 minutes). Participants were introduced to the scope of the project and given time to get used to the HoloLens. Then, the instructor explained the four core interactions: 3D drawing, 2D drawing, *canvas* creation, *canvas* manipulation, and other helper tools. Participants practiced using the above tools until they felt confident. The workspace scaling features were not used in the study.

Fixed tasks (30-45 minutes). We designed two fixed tasks to compare SymbiosisSketch to existing 3D sketching systems. We counterbalanced the order of the systems amongst the participants to eliminate learning effects.

The first task involved drawing the outline of a bullet shaped building with textured features (Figure 15a). Participants had to perform the task using two systems: *symbiosis*, and *mid-air only*, which did not afford *canvas* creation and stroke projection, therefore replicating existing mid-air drawing systems such as Tilt Brush. For the former, participants were encouraged to utilize the *drawing canvas* tool.

Task-2 involved drawing a logo over a curved physical surface (see Figures 15b-c). Participants were encouraged to rest the tip of their pen on the physical surface. The motivation behind the second task was to measure the usefulness of the 2D interface even when the user can get better motor control by resting on a physical surface.

Expanded overview & exploration (5-10 minutes). After completing the fixed tasks, participants were told about solid surfaces, workspace scaling, and the *canvas* bookmarking feature.

Freeform exploration and design (30-45 minutes). Finally, participants were free to explore the system on their own to create designs in the fourth phase. They were provided a host of physical props to inspire their designs. They completed a questionnaire after the study.

Results for Fixed Tasks

For the fixed tasks, we evaluated the designs using two quantitative measures: *task completion time* and *total strokes drawn*, defined simply as the number of strokes in the final design. The latter provides a proxy measure for the level of detail present in the designs.

For each participant, the *task completion time*, as well as the *total strokes drawn*, were higher when using the *symbiosis* system. In task-1, mean task completion time in the *symbiosis* condition ($m = 519s, \sigma = 115s$) was over two times that of *mid-air only* system ($m = 242s, \sigma = 122s$). A similar trend was observed for the second task: ($m = 281s, \sigma = 123s$) vs ($m = 148s, \sigma = 85s$). Users using the combined 2D/3D interface of *symbiosis* drew 75% more strokes ($m = 64, \sigma = 24$).

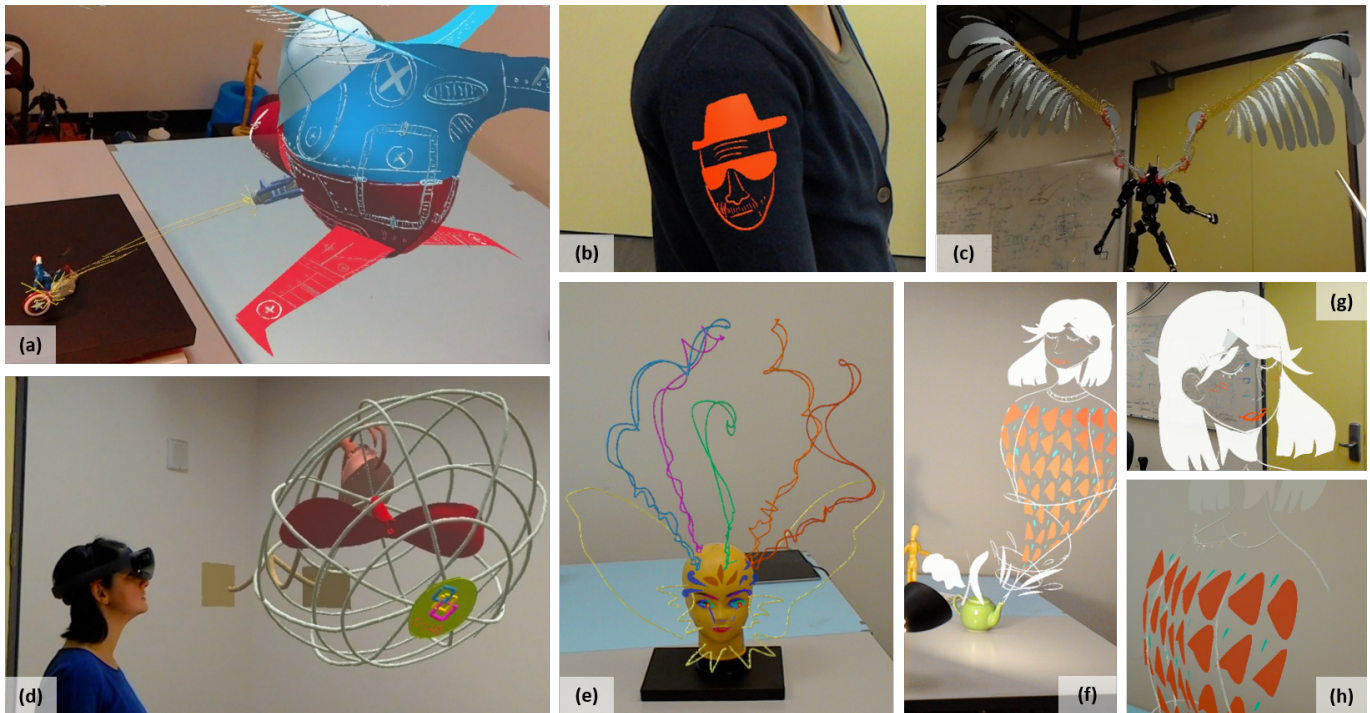


Figure 14. Sample results showing SymbiosisSketch's creative potential. Author creations: war helicopter shooting at a Captain America figure (a), logo drawn on a person's clothing (b), mechanical wing augmenting a robot figure (c), and a large wall-fan (d). Participant creations: scary figure utilizing a mannequin head (e) and genie emanating out of a teapot (f) with details (g, h).

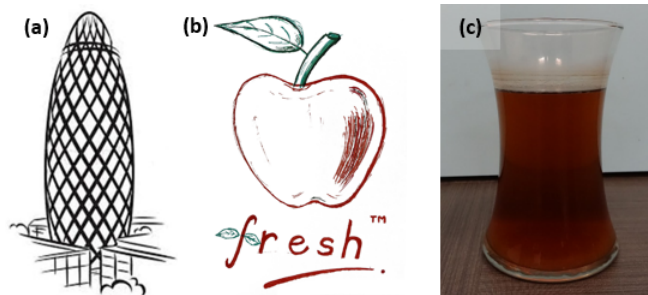


Figure 15. Task-1: drawing London's Gherkin building (a). Task-2: drawing a logo (b) on a curved physical surface of a vase (c).



Figure 16. Representative results for the fixed tasks. Task 1: Gherkin building, using *symbiosis* (left) and *mid-air only* (center-left). Task 2: logo on physical object, *symbiosis* (center-right), *mid-air only* (right).

than *mid-air only* ($m = 37, \sigma = 14$), for the first task. This difference increased to 140% for the texture detail-heavy second task: ($m = 67, \sigma = 38$) vs ($m = 28, \sigma = 20$). Paired t-tests reveal that the two conditions were statistically significant ($p < .05$) on the task completion time measure for both tasks, and the total strokes drawn measure for task-2. However, these statistics should be considered with caution since our focus was on qualitative measures meant that the tasks were not carefully controlled.

Our observations during the study suggest that these results can be explained by two main factors. First, participants took some time getting used to the *canvas* creation tool, which required more mental effort than *mid-air* sketching. However, once participants understood how to use the tool, they were able to utilize it effectively, and draw detailed designs more accurately when they had access to the 2D UI. Figures 16, 18 qualitatively

demonstrate that the *symbiosis* condition produced drawings that were more precise, and better-reproduced details shown in the target images.

Freeform Design and Qualitative Feedback

Participants were successful in utilizing SymbiosisSketch for a variety of design tasks. During the brief design sessions, participants attempted a design sketch of a car (P2), environment design involving physical props (P4, P6), character augmentations (P3, P4), and mystical art (P5). Figure 14 shows some of the resulting artifacts.

Overall, the qualitative feedback collected indicated that participants found SymbiosisSketch's toolset useful. This is evident in P5's comment that the system is "*a great way to bridge old methods with new technology*", but "[*composing*] a 3D image

will require a lot of work and planning”. Below, we report the feedback received for individual components.

Drawing on Tablet to Create 3D Sketches

Participants found this feature to be extremely useful. On a 5-point Likert scale (1=not useful, 5=extremely useful), the median score was 4.5 for usefulness and 4.0 for usability. They appreciated the degree of control in traditional sketching, which they could replicate in 3D sketches. Mapping a drawing surface to physical objects in the world was judged to be a useful and enjoyable feature.

P3: “Drawing in 3D can feel unhinged and out of control, and this capability made me feel some of the enjoyment and control I feel when I practice traditional drawing.”

Ability to Combine 2D and 3D Sketching in One Design

Participants responded positively to the multimodal interface for 3D design. They found the hybrid interface very useful (rated 4.5 out of 5), as well as easy to use (rated 4 out of 5).

P5: “I really liked using the planes, and the combination of 2D & 3D. I really like the consideration of how working solely in a 3D space can become hard for most people.”

An exception was *P2*, who, used to ZBrush [3] sculpting, finds freeform 3D sketching/modelling challenging.

Curved Canvas Creation and Stroke Projection

Users unanimously found the ability to create *drawing canvases* very useful (rated 5 out of 5).

P4: “Mapping of the canvas in 3D was what I liked the most. Once I had defined a filled, curved region that I was happy with I found that it was very enjoyable to draw curves starting from that surface.”

However, some participants thought the feature was difficult to use (median usability 3) due to control issues. In the future, we would like to add finer curvature control to the *curved canvas* tool to address these issues.

Use of Physical Props and Surfaces

Use of physical props was assessed to be both useful (median 4.5) and easy to use (median 4). In fact, all participants utilized this feature, and included physical objects in their creations. They also traced these objects to define *canvases*.

P1: “Mapping a drawing surface to physical objects in the world was an immensely useful and enjoyable feature.”

Miscellaneous Feedback

Participants also provided us feedback on other aspects of the system. In particular, they found *solid surfaces* to be very useful and aesthetically pleasing. Participants (*P1*, *P3*, *P4*) also wanted non-planar closed curves drawn mid-air to be useable to trim a surface. This is an interesting technical challenge for future work: a possible implementation could project the curve onto the best-fit plane and then lift a 2D triangulation to 3D. Finally, *P2* suggested an additional interaction mode: drawing on the tablet and generating a swept surface by moving it in 3D. We imagine this to be similar to, but more versatile than, Surface Drawing [45]. Controlling the tablet’s motion and orientation mid-air, however, may be challenging.

DISCUSSION

Our evaluation suggested that participants enjoyed our core ideas, and were able to utilize the system successfully for creating the designs they desired. The artifacts created by the study participants and those created by the authors, confirm that SymbiosisSketch effectively mixes the traditional with the new to create a compelling and novel hybrid interface for 3D concept design. Along with the encouraging results, the user evaluation also showed some limitations, with opportunities to improve in the future.

Limitations and Future work

Drawing Canvases

Participants (*P1*, *P4*) expressed the desire to go beyond current surface patch topology, to create surfaces with multiple boundaries like open cylinders, or closed surfaces like spheres. Fitting a general surface poses new problems: inferring surface topology, defining a non-trivial mapping to a *drawing canvas*, and the need for view manipulation in the tablet to draw on such a *canvas*.

In Situ Design in the Wild and Other Hardware Limitations

P3 expressed the desire to using our system to plan and create large murals. Our use of a motion capture system to track the 3D stylus makes our current setup impractical outside a lab. The small field of view on the HoloLens also made it hard to view large strokes in totality (*P6*). Better 3D acquisition of the physical environment in AR can allow complex surfaces to become *drawing canvases*, but in turn poses the problem of scene surface segmentation.

Stroke & Canvas Manipulation for Iterative Design

While our smoothing method effectively removes tracking noise and motor control error from 3D strokes, it cannot remove *errors of intent*—caused by conflicting visual percepts or users changing their mind later. Non-rigid manipulation of *drawing canvases* and correction of freeform 3D curves by tablet-based oversketching are great avenues for future exploration. Effective manipulation and editing tools can improve the value of our system for the iterative design process.

CONCLUSION

SymbiosisSketch is a hybrid system aiming to blend design knowledge and expertise from the traditional 2D sketching domain to the new and exciting world of direct 3D sketching in AR/VR. By devising a novel method for defining constrained *drawing canvases* in 3D and building tools to support interaction with physical objects, we have built a system for 3D conceptual design in situ. Our user evaluation confirmed that our toolset is useful, effective, and is able to support a variety of design tasks for users with diverse artistic backgrounds. We hope that this work guides future research in visual communication, furthering our ability to efficiently transform mental design concepts into digital models.

ACKNOWLEDGEMENTS

We thank Michael Tao, Nicole Sultanum, Sarah Kushner, and Derek Liu for helping us shoot various video clips. We also sincerely thank our study participants for their time, effort, and patience.

REFERENCES

1. 2017. Gravity Sketch - Bringing Virtual Reality Into Your Design Workflow. (2017).
<https://www.gravitysketch.com/>
2. 2017. MixedRealityToolkit-Unity. (Sept. 2017). <https://github.com/Microsoft/MixedRealityToolkit-Unity>
original-date: 2016-01-28T18:54:58Z.
3. 2017. Pixologic : ZBrush - The all-in-one-digital sculpting solution. (2017). <http://pixologic.com>
4. 2017. Tilt Brush by Google. (2017).
<https://www.tiltbrush.com/>
5. Fatemeh Abbasinejad, Pushkar Joshi, and Nina Amenta. 2011. Surface Patches from Unorganized Space Curves. *Computer Graphics Forum* 30, 5 (Aug. 2011), 1379–1387. DOI:
<http://dx.doi.org/10.1111/j.1467-8659.2011.02012.x>
6. Ian G. Angus and Henry A. Sowizral. 1995. Embedding the 2D interaction metaphor in a real 3D virtual environment, Vol. 2409. International Society for Optics and Photonics, 282–294. DOI:
<http://dx.doi.org/10.1117/12.205875>
7. Rahul Arora, Rubaiat Habib Kazi, Fraser Anderson, Tovi Grossman, Karan Singh, and George Fitzmaurice. 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, New York, NY, USA, 5643–5654. DOI:
<http://dx.doi.org/10.1145/3025453.3025474>
8. Seok-Hyung Bae, Ravin Balakrishnan, and Karan Singh. 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*. ACM, 151–160. DOI:
<http://dx.doi.org/10.1145/1449715.1449740>
9. Seok-Hyung Bae, Ravin Balakrishnan, and Karan Singh. 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, New York, NY, USA, 59–68. DOI:
<http://dx.doi.org/10.1145/1622176.1622189>
10. Katie Bassett, Ilya Baran, Johannes Schmid, Markus Gross, and Robert W. Sumner. 2013. Authoring and Animating Painterly Characters. *ACM Trans. Graph.* 32, 5, Article 156 (Oct. 2013), 12 pages. DOI:
<http://dx.doi.org/10.1145/2484238>
11. Mikhail Bessmeltsev, Caoyu Wang, Alla Sheffer, and Karan Singh. 2012. Design-driven quadrangulation of closed 3D curves. *ACM Transactions on Graphics* 31, 6 (Nov. 2012), 1. DOI:
<http://dx.doi.org/10.1145/2366145.2366197>
12. Richard L. Bishop. 1975. There is More than One Way to Frame a Curve. *The American Mathematical Monthly* 82, 3 (March 1975), 246. DOI:
<http://dx.doi.org/10.2307/2319846>
13. D. A. Bowman, J. Wineman, L. F. Hodges, and D. Allison. 1998. Designing animal habitats within an immersive VE. *IEEE Computer Graphics and Applications* 18, 5 (Sept. 1998), 9–13. DOI:
<http://dx.doi.org/10.1109/38.708555>
14. D. A. Bowman and C. A. Wingrave. 2001. Design and evaluation of menu systems for immersive virtual environments. In *Proceedings IEEE Virtual Reality 2001*. 149–156. DOI:
<http://dx.doi.org/10.1109/VR.2001.913781>
15. Xuejin Chen, Sing Bing Kang, Ying-Qing Xu, Julie Dorsey, and Heung-Yeung Shum. 2008. Sketching Reality: Realistic Interpretation of Architectural Designs. *ACM Trans. Graph.* 27, 2 (May 2008), 11:1–11:15. DOI:
<http://dx.doi.org/10.1145/1356682.1356684>
16. Xiang Anthony Chen, Jeeun Kim, Jennifer Mankoff, Tovi Grossman, Stelian Coros, and Scott E. Hudson. 2016. Reprise: A Design Tool for Specifying, Generating, and Customizing 3D Printable Adaptations on Everyday Objects. ACM Press, 29–39. DOI:
<http://dx.doi.org/10.1145/2984511.2984512>
17. Kelvin Cheng and Masahiro Takatsuk. 2009. Interaction Paradigms for Bare-Hand Interaction with Large Displays at a Distance. In *Human-Computer Interaction*, Inaki Maurtua (Ed.). InTech, 195–222. DOI: 10.5772/7733.
18. Siu-Wing Cheng, Tamal Krishna Dey, and Jonathan Richard Shewchuk. 2013. *Algorithms for constructing Delaunay triangulations*. CRC Press / Taylor & Francis, Boca Raton, Fla. OCLC: 844619099.
19. Zeynep Cipiloglu, Abdullah Bulbul, and Tolga Capin. 2010. A framework for enhancing depth perception in computer graphics. ACM Press, 141. DOI:
<http://dx.doi.org/10.1145/1836248.1836276>
20. Josh Constine. 2016. Oculus “Quill” Turns VR Painting Into Performance Art. (2016).
<http://social.techcrunch.com/2016/01/26/oculus-quill/>
21. Chris De Paoli and Karan Singh. 2015. SecondSkin: Sketch-based Construction of Layered 3D Models. *ACM Trans. Graph.* 34, 4 (July 2015), 126:1–126:10. DOI:
<http://dx.doi.org/10.1145/2766948>
22. Michael F. Deering. 1995. HoloSketch: A Virtual Reality Sketching/Animation Tool. *ACM Trans. Comput.-Hum. Interact.* 2, 3 (Sept. 1995), 220–238. DOI:
<http://dx.doi.org/10.1145/210079.210087>
23. Gianluca Donato and Serge Belongie. 2002. Approximate Thin Plate Spline Mappings. In *Computer Vision âĖ ECCV 2002*, Gerhard Goos, Juris Hartmanis, Jan van Leeuwen, Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen (Eds.). Vol. 2352. Springer Berlin Heidelberg, Berlin, Heidelberg, 21–31.
http://link.springer.com/10.1007/3-540-47977-5_2 DOI: 10.1007/3-540-47977-5_2.

24. Tomãas Dorta, Gokce Kinayoglu, and Michael Hoffmann. 2016. Hyve-3D and the 3D Cursor: Architectural co-design with freedom in Virtual Reality. *International Journal of Architectural Computing* 14, 2 (2016), 87–102. DOI: <http://dx.doi.org/10.1177/1478077116638921>
25. David Eberly. 2000. Least squares fitting of data. (2000).
26. Koos Eissen and Roselien Steur. 2011. *Sketching: the basics ; the prequel to Sketching: drawing techniques for product designers*. BIS, Amsterdam. OCLC: 756275344.
27. Michele Fiorentino, Giuseppe Monno, Pietro Alexander Renzulli, and Antonio E. Uva. 2003. 3D Sketch Stroke Segmentation and Fitting in Virtual Reality. In *International Conference Graphicon*.
28. Tovi Grossman, Ravin Balakrishnan, Gordon Kurtenbach, George Fitzmaurice, Azam Khan, and Bill Buxton. 2002. Creating Principal 3D Curves with Digital Tape Drawing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '02)*. ACM, New York, NY, USA, 121–128. DOI: <http://dx.doi.org/10.1145/503376.503398>
29. Andrew J. Hanson and Hui Ma. 1995. *Parallel transport approach to curve framing*. Technical Report TR425. 4–7 pages. <ftp://ftp.cs.indiana.edu/pub/hanson/iucs-tr425.ps>
30. Ian P. Howard and Brian J. Rogers. 2008. Depth from monocular cues and vergence. In *Seeing in Depth*. Oxford University Press, 355–410. <http://www.oxfordscholarship.com/view/10.1093/acprof:oso/9780195367607.001.0001/acprof-9780195367607> DOI: 10.1093/acprof:oso/9780195367607.001.0001.
31. Kiia Kallio. 2005. 3D6B Editor: Projective 3D Sketching with Line-Based Rendering. The Eurographics Association. <http://diglib.eg.org/handle/10.2312/SBM.SBM05.073-079> DOI: 10.2312/SBM/SBM05/073-079.
32. L. B. Kara and K. Shimada. 2007. Sketch-Based 3D-Shape Creation for Industrial Styling Design. *IEEE Computer Graphics and Applications* 27, 1 (Jan. 2007), 60–71. DOI: <http://dx.doi.org/10.1109/MCG.2007.18>
33. Rubaiat Habib Kazi, Tovi Grossman, Hyunmin Cheong, Ali Hashemi, and George Fitzmaurice. 2017. DreamSketch: Early Stage 3D Design Explorations with Sketching and Generative Design. ACM, Quebec City, QC, Canada.
34. Daniel Keefe, Robert Zeleznik, and David Laidlaw. 2007. Drawing on air: Input techniques for controlled 3D line illustration. *IEEE Transactions on Visualization and Computer Graphics* 13, 5 (2007), 1067–1081. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4135646
35. Daniel F. Keefe, Daniel Acevedo Feliz, Tomer Moscovich, David H. Laidlaw, and Joseph J. LaViola Jr. 2001. CavePainting: a fully immersive 3D artistic medium and interactive experience. In *Proceedings of the 2001 symposium on Interactive 3D graphics*. ACM, 85–93. DOI: <http://dx.doi.org/10.1145/364338.364370>
36. Yongkwan Kim and Seok-Hyung Bae. 2016. SketchingWithHands: 3D Sketching Handheld Products with First-Person Hand Posture. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 797–808. DOI: <http://dx.doi.org/10.1145/2984511.2984567>
37. Vojtěch Krs, Ersin Yumer, Nathan Carr, Bedrich Benes, and Radomír Měch. 2017. Skippy: single view 3D curve interactive modeling. *ACM Transactions on Graphics* 36, 4 (July 2017), 1–12. DOI: <http://dx.doi.org/10.1145/3072959.3073603>
38. P. Lapides, E. Sharlin, M.C. Sousa, and L. Streit. 2006. The 3D Tractus: A Three-Dimensional Drawing Board. *IEEE*, 169–176. DOI: <http://dx.doi.org/10.1109/TABLETOP.2006.33>
39. Yuwei Li, Xi Luo, Youyi Zheng, Pengfei Xu, and Hangbo Fu. 2017. SweepCanvas: Sketch-based 3D Prototyping on an RGB-D Image. ACM, Quebec City, QC, Canada.
40. John Finley Lucas. 2005. *Design and evaluation of 3D multiple object selection techniques*. Ph.D. Dissertation. Virginia Tech.
41. Patrick Paczkowski, Min H. Kim, Yann Morvan, Julie Dorsey, Holly Rushmeier, and Carol O'Sullivan. 2011. Insitu: Sketching Architectural Designs in Context. In *Proceedings of the 2011 SIGGRAPH Asia Conference (SA '11)*. ACM, New York, NY, USA, 182:1–182:10. DOI: <http://dx.doi.org/10.1145/2024156.2024216>
42. Scott Robertson and Thomas Bertling. 2013. *How to draw: drawing and sketching objects and environments from your imagination*. Designstudio Press, Culver City, Calif. OCLC: 897060023.
43. E. Sachs, A. Roberts, and D. Stoops. 1991. 3-Draw: a tool for designing 3D shapes. *IEEE Computer Graphics and Applications* 11, 6 (Nov. 1991), 18–26. DOI: <http://dx.doi.org/10.1109/38.103389>
44. Bardia Sadri and Karan Singh. 2014. Flow-complex-based Shape Reconstruction from 3D Curves. *ACM Trans. Graph.* 33, 2 (April 2014), 20:1–20:15. DOI: <http://dx.doi.org/10.1145/2560328>
45. Steven Schkolne, Michael Pruett, and Peter Schröder. 2001. Surface Drawing: Creating Organic 3D Shapes with the Hand and Tangible Tools. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '01)*. ACM, New York, NY, USA, 261–268. DOI: <http://dx.doi.org/10.1145/365024.365114>
46. Ryan Schmidt, Azam Khan, Karan Singh, and Gord Kurtenbach. 2009. Analytic drawing of 3D scaffolds. ACM Press, 1. DOI: <http://dx.doi.org/10.1145/1661412.1618495>

47. Ryan Schmidt and Karan Singh. 2008. Sketch-Based Procedural Surface Modeling and Compositing Using Surface Trees. *Computer Graphics Forum* 27, 2 (April 2008), 321–330. DOI: <http://dx.doi.org/10.1111/j.1467-8659.2008.01129.x>
48. Karan Singh. 2006. Industrial motivation for interactive shape modeling: a case study in conceptual automotive design. ACM Press, 3. DOI: <http://dx.doi.org/10.1145/1185657.1185671>
49. Ivan E. Sutherland. 1963. Sketchpad: a man-machine graphical communication system. ACM Press, 329. DOI: <http://dx.doi.org/10.1145/1461551.1461591>
50. Zsolt Szalavári and Michael Gervautz. 1997. The Personal Interaction Panel—a Two-Handed Interface for Augmented Reality. *Computer Graphics Forum* 16, 3 (Sept. 1997), C335–C346. DOI: <http://dx.doi.org/10.1111/1467-8659.00137>
51. Shun'ichi Tano, Shinya Yamamoto, Junko Ichino, Tomonori Hashiyama, and Mitsuru Iwata. 2013. Truly Useful 3D Drawing System for Professional Designer by “Life-Sized and Operable” Feature and New Interaction. In *Human-Computer Interaction—INTERACT 2013*. Number 8117 in Lecture Notes in Computer Science. Springer Berlin Heidelberg, 37–55. DOI: 10.1007/978-3-642-40483-2_3.
52. Yannick Thiel, Karan Singh, and Ravin Balakrishnan. 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*. ACM, 383–392. DOI: <http://dx.doi.org/10.1145/2047196.2047246>
53. Gerold Wesche and Hans-Peter Seidel. 2001. FreeDrawer: a free-form sketching system on the responsive workbench. ACM Press, 167. DOI: <http://dx.doi.org/10.1145/505008.505041>
54. Brian Whited, Eric Daniels, Michael Kaschalk, Patrick Osborne, and Kyle Odermatt. 2012. Computer-assisted Animation of Line and Paint in Disney’s Paperman. In *ACM SIGGRAPH 2012 Talks (SIGGRAPH ’12)*. ACM, New York, NY, USA, Article 19, 1 pages. DOI: <http://dx.doi.org/10.1145/2343045.2343071>
55. Baoxuan Xu, William Chang, Alla Sheffer, Adrien Bousseau, James McCrae, and Karan Singh. 2014. True2Form: 3D Curve Networks from 2D Sketches via Selective Regularization. *ACM Trans. Graph.* 33, 4 (July 2014), 131:1–131:13. DOI: <http://dx.doi.org/10.1145/2601097.2601128>
56. Maria C Yang and Jorge G Cham. 2007. An analysis of sketching skill and its role in early stage engineering design. *Journal of Mechanical Design* 129, 5 (2007), 476–482.
57. Youyi Zheng, Han Liu, Julie Dorsey, and Niloy J. Mitra. 2016. SmartCanvas: Context-inferred Interpretation of Sketches for Preparatory Design Studies. In *Computer Graphics Forum*, Vol. 35. Wiley Online Library, 37–48.

Appendix A: Additional Results



Figure 17. (Clockwise from top-left) close-ups of selected results showing details of designs: mechanical wing (Figure 14c), war helicopter (Figure 14a), large wall-fan (Figure 14d), and some additional results: *CHInosaur* as the Toronto Raptor (participant creation) and skirt drawn over a human model (author creation).

Appendix B: Fixed Study Task Results



Figure 18. All participants' drawings for the fixed tasks. Row-wise from top: task 1 (Gherkin building) in *symbiosis* condition, the in *mid-air only* condition; task 2 (logo on a physical surface) in *symbiosis* condition, and in *mid-air only* condition.