Creative Expression with Immersive 3D Interactions

Rahul Arora
University of Toronto
Toronto, ON M5S 2E4, Canada
arorar@dgp.toronto.edu

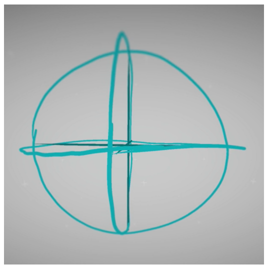
CHI '20 Extended Abstracts, April 25–30, 2020, Honolulu, HI, USA.

ACM ISBN 978-1-4503-6819-3/20/04.
http://dx.doi.org/10.1145/3334480.3375028

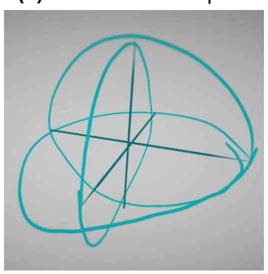
Abstract
Virtual and augmented realities (VR/AR) allow artists to create 3D content in a three-dimensional space—both display and inputs are 3D. Getting rid of 2D proxies such as screens and graphic tablets removes a significant barrier from 3D creation and allows artists to create more intuitively, and potentially more efficiently. However, creating in VR/AR introduces new control, precision, and ergonomic challenges. Designing interactive tools for 3D creation is therefore non-trivial. A deep understanding of human factors, user preferences, as well as biases stemming from users' experience with 2D tools is essential to develop effective creative tools for VR/AR. My research combines exploratory user studies and technical advancements to build novel tools for creating 3D content in immersive spaces.

I present two computer graphics applications which utilize 3D interactions to improve existing creative workflows and devise novel ones for visual creative expression in three-dimensions. The first studies concept sketching, while the second explores animation of dynamic physical phenomena. I then describe my ongoing work and planned future work on other creative applications.

Author Keywords
immersive reality; virtual and augmented realities; 3D interaction; creative tools



(a) The artist's viewpoint.



(b) An alternate viewpoint.

Figure 1: Drawing accurately in three-dimensions can be difficult. Sketches which look perfect from one viewpoint (top) can show massive inaccuracies when inspected from another angle (bottom).

CCS Concepts

•Human-centered computing → Interaction techniques; *Mixed / augmented reality; Virtual reality;* **•Computing methodologies → Computer graphics;**

Introduction

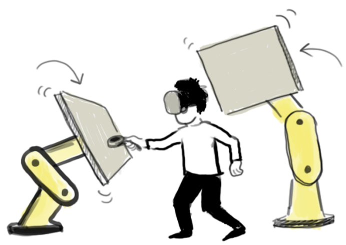
Humans have enjoyed artistic expression in the form of three-dimensional artifacts for millennia. Starting with pottery and statue sculpting [7], 3D creative expression now takes many digital forms such as virtual characters, 3D games, animated films, and 3D modelling for fabrication. Traditionally, digital creation is performed using two-dimensional proxies—2D input devices such as mice and tablets as well as 2D display technologies. However, advancements in immersive displays and 3D tracking have opened up the third dimension for creative expression. Immersive spaces such as virtual and augmented realities (VR/AR) are allowing end-users to create directly in 3D; and our digital devices are finally mimicking the physical tools we've used for generations. My research aims to build intelligent tools for 3D media creation by developing novel interaction mechanisms.

Immersive reality is, however, not a fundamentally new technology. Immersive VR traces its roots to Ivan Sutherland's [22] invention of a tethered head-mounted display (HMD). Headset-based AR also dates back to the 90s, with Rosenberg's Virtual Fixtures [21] offering perhaps the first working implementation. Unfortunately, early hardware offerings were typically too constrained to allow for any meaningful creative use of VR/AR—tethered HMDs, limited tracking, tiny fields of view, and low-resolution displays seriously encumbered artists willing to create in immersive reality. The new renaissance of VR/AR is driven by technical advancements in both display and tracking technologies [6]. With rapidly maturing technology, commercial interest [12], mass-manufacturing, and consumer-friendly prices, VR and AR are promising to write the next-big success story in consumer hardware. The time is thus ripe to explore serious applications for immersive reality.

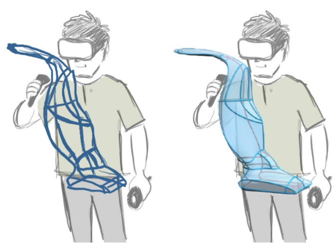
My personal interest is in computational support for the creation of visual media. During my master's degree, I explored computational tools for concept design using traditional (2D) sketching [2]. While starting my PhD research at the University of Toronto, I was drawn towards the novel possibilities afforded by 3D creation and started exploring creative applications of immersive realities. My dissertation aims to understand not just the benefits of immersion, but its fundamental limitations as well. While building immersive creation tools, I borrow from the extensive body of work on desktop-based tools, recognizing how physical and virtual constraints can help improve creative workflows in 3D. The hope is that by harmoniously combining existing desktop-based workflows with novel design capabilities enabled by VR/AR, my tools can help artists create truly unique artifacts which were either too cumbersome or completely impossible to produce using existing tools.

Experimental Evaluation of 3D Sketching

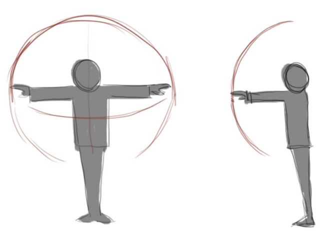
Sketching is a prized conceptualization tool due to its freeform and expressive nature. Naturally, the first creative application I explored was the use of virtual reality for drawing mid-air—3D sketching. Prior to my own work on 3D sketching, a number of HCI researchers [15, 24, 14] had explored 3D sketching interfaces for conceptual art and design. Researchers had explored if artists can get better at 3D drawing with experience [24] and whether force-feedback can help sketch precisely [15]. These works indicated that designers enjoy the unbridled freedom of drawing mid-air, but were frustrated by their inability to draw precisely. To better understand how and why 3D sketches tend to be imprecise compared to their 2D counterparts, I decided to perform a structured exploration of the various physical, visual, and ergonomic factors impact 3D sketching ability.



(a) Utilize physical surfaces for improved precision.



(b) Provide visual guidance to improve depth perception.



(c) Emphasize *sketchable* surface orientations.

Figure 2: Design guidelines for VR-based sketching tools.

Preliminary experiments performed by my collaborators and I suggested that even when drawing mid-air, concept sketching would benefit from the presence of constraints. Rarely did we find ourselves drawing arbitrary 3D strokes; most strokes would ideally be anchored onto *imaginary surfaces* that we wanted to depict, and to fixed points in space, often onto existing strokes in the sketch (Figure 1). Armed with this anecdotal experiment, we invited five professional artists for semi-structured sessions with Tilt Brush—a commercial tool for freeform 3D sketching—followed by interviews. These observational sessions, while reaffirming the instinctive nature of drawing directly in three-dimensions, suggested three major themes of struggle.

– *Importance of precision.* Stroke precision was deemed to be extremely important in 3D, even more so than in 2D, since the viewer does not need to take a mental leap from 2D to 3D when looking at a sketch executed in 3D.
– *Ergonomic factors.* Strokes often meant to be straight or planar did not end up being so since it was hard to maintain a straight line.
– *Visual factors.* Precise drawing was difficult since perceiving the correct depth to position the pen was inconvenient.

With these observations in mind, I designed two quantitative studies on VR-sketching.

Comparing Traditional and VR Sketching
A major difference between drawing in 3D vs in 2D is the lack of physical constraint provided by the drawing plane. However, when drawing in VR, another distinguishing factor is the reduced visual fidelity of the virtual world. In order to isolate the impact of these two factors, I set up a study where participants drew planar strokes on a physical plane (*traditional* condition), on the same plane but when wearing a VR HMD (*hybrid* condition), and drawing mid-air in VR (*VR* condition). Another distinguishing feature of mid-air drawing is the ability to draw in arbitrary orientations. To gauge the ergonomic issues encountered when utilizing different drawing plane orientations, participants drew these shapes on a horizontal plane, a frontal vertical plane facing them, and a "sideways" vertical plane perpendicular to the frontal plane.

The study revealed that the lack of physical constraint was indeed the major driver of VR-sketching imprecision. Further, the unusually-oriented "sideways" plane proved to be cumbersome to draw on. A subsequent experiment dug further into the factors influencing mid-air drawing, including a deeper exploration into the orientation of drawing surfaces.

Factors Affecting Mid-Air Drawing
The second study aimed to understand how 3D sketching performance could be improved by providing non-constraining visual guidance, such as rendering the drawing surface or the target stroke itself. Following initial observations that strokes in a concept sketch are almost always meant to depict surfaces, this experiment also compared planar drawing performance to drawing strokes on curved surfaces.

An interesting observation was that providing higher levels of visual guidance helped improved sketching accuracy, but stroke quality, measured via fairness, suffered.

Design Guidelines for VR-Sketching Tools
I filtered the results of the observational study and the two quantitative experiments into a succinct set of design guidelines for VR-sketching tools (Figure 2). The guidelines can help designers of future 3D sketching tools build intuitive interfaces which help users draw efficiently, with confidence and precision. I then built one such tool myself—combining 2D and 3D sketching in a coherent interface.

SymbiosisSketch: Combining 2D and 3D

The goals of the SymbiosisSketch project were twofold—one was simply the exploration and implementation of the design guidelines inferred from my VR-sketching experiments, and the other was to bring the expressiveness and stylistic richness of 2D sketches [11] to the 3D world. SymbiosisSketch [3] is an AR-based sketching tool. Users utilize a HoloLens HMD and a graphic tablet, along with a single drawing pen that allows drawing both 3D strokes mid-air and surface-constrained strokes using the tablet (Figure 3). While mid-air drawing aids depth depiction and creation of freeform geometry, 2D strokes help create fine details, textures, and well-known artistic styles. In addition to mid-air drawing, the tool implements a number of novel interaction techniques, utilizing metaphors described below.

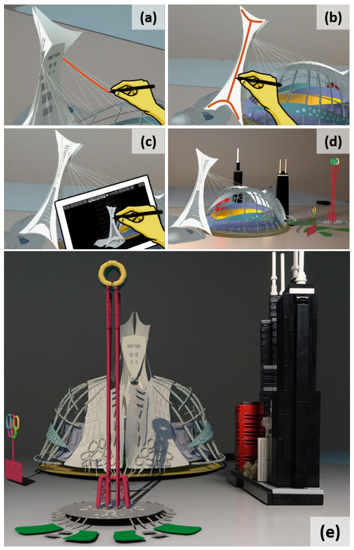


Figure 3: SymbiosisSketch combines tablet-bound sketching (a) with mid-air strokes (b). Users can fit *canvases* to curves and use a tablet to project strokes onto them (c). Designing in situ places physical and virtual objects in the same sketch (d), which can be post-processed to create a coherent look (e).

Planar and Curved Drawing Canvases

In order to allow depicting non-planar surfaces with rich details, SymbiosisSketch introduces the concept of *drawing canvases*. A *drawing canvas* is a surface a user can draw on using the tablet. Creating and managing these *canvases* is easy—users can draw a few curves in the air and a best-fit surface is automatically computed (Figure 3b). The *canvases* can be transformed using standard translation, rotation, and scaling widgets in 3D. In particular, these transformations allow the creation of extruded details and architectural embellishments efficiently (Figure 3e).

Projected Drawing and Solid Surfaces

When a *drawing canvas* is selected, 2D tablet-bound strokes are mapped to the selected canvas (Figure 3c). The tablet does not need to be physically collocated with the *canvas*, and the user is free to position the tablet in the most comfortable position and orientation. This ergonomically-optimal indirect mapping is in contrast to existing systems where the tablet is tracked in three-dimensions [10] and interactions on the tablet are directly mapped to 3D. Three-dimensional positioning is aided by showing a virtual pointer at the spatial location of the projected pen position, including when the pen is not actively drawing but is simply hovering over the tablet. The use of see-through AR (instead of VR) is extremely helpful here—especially for drawing small details and hatching—as the user is able to utilize the full-resolution of the tablet. Current VR display technologies do not offer resolutions anywhere close to professional drawing tablets. Finally, a *fill* tool creates solid surfaces, improving a viewer's depth perception by the way of occlusion, lighting, and shadows.

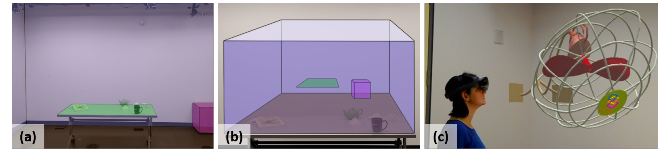


Figure 4: Automatic plane detection (a) helps anchor strokes onto physical features. Workspace scaling (b) creates a bird's eye view of the world. A combination of the two helps create room-scale objects positioned accurately in the real-world (c).

Interacting with Physical Objects

A key advantage of AR is drawing *in situ*, i.e., creating artwork incorporating the real world (Figure 3d,e). A planar surface detection tool helps users in this task by automatically detecting planes in the physical world—such as tables, walls, and floors—and snapping *planar drawing canvases* onto them (Figure 4a). Experiments showed that even when drawing over physical surfaces, users draw more accurately when using the indirect tablet-*canvas* mapping than drawing directly over the physical surface in 3D. Lastly, a *workspace scaling* tool aided the visualization of room-scale drawings and the creation of very large or very small features by "zooming in and out" of the real-world scale (Figure 4b,c).

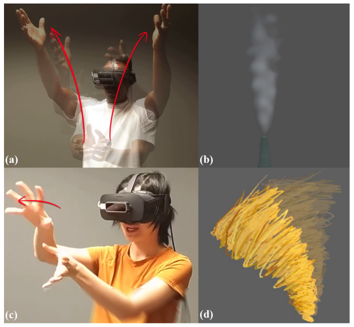


Figure 5: Two examples gestures from the gesture-based animation formative study. A high bandwidth gesture simultaneously controls the direction, spread, and turbulence of a smoke simulation (a–b).A direct manipulation gesture bends a tornado to depict a follow-through effect (c–d).

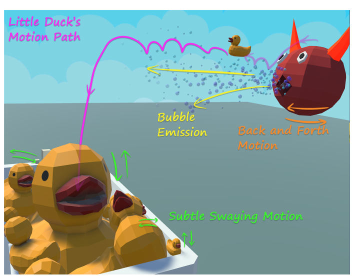


Figure 6: An animation created using the MagicalHands system, utilizing directly performed as well as keyframed transformations, and animator-controlled particle systems.

Hand Gestures for VR-Animation

In addition to the interest in 3D sketching, the rise in VR's popularity has also given rise a number of commercial animation tools [18, 23]. While these tools are intuitive to get started with and fun to play with, they do not utilize the full power of immersive interaction capabilities. Specifically, these tools do not take advantage of the enhanced possibility of embodied interaction, and do not allow users to freely mix between different modes of animation authoring. For example, Quill [18] is a frame-by-frame animation tool and does not allow animators to control objects semantically—there is no semantic relationship between objects on different frames of the animation[1].

Animation authoring involves the creation and manipulation of complex spatiotemporal phenomena, interplay of numerous physical attributes and abstract phenomena, and hierarchical relationships between objects. Hand gestures are perfectly suited for such complex tasks; not only does the versatility of hand poses help manipulate many animation controls efficiently, but interacting directly with virtual objects with their hands improves users' immersion in VR. A large body of work in computer graphics (see, for example, [16, 9]) has explored performance-based animation of characters using hand gestures. In contrast, I decided to focus on the animation of dynamic physical phenomena such as smoke, rain, fractures, and rigid-body interactions. Compared to the mapping between the motion of an animator's hands and the resulting movement of an animated character, the relationship between gestures and physically-based animations is much harder to establish. As a result, I decided to conduct a gesture-elicitation study to understand the use of mid-air hand gestures for animating in VR [4].

[1] Introduction of a timeline in Quill 2.0 (released Aug 1, 2019) somewhat alleviates this problem.

Gesture Elicitation Study

While the elicitation study is inspired from Wobbrock et al.'s groundbreaking work on participatory design [25] and similar studies on mid-air hand gestures (for example, [19]), there is an important distinction. Most existing studies prompted the user with a standard task, and seek the most commonly utilized gestures for that task. Given the diversity of animated phenomena I was targeting, it was unclear what a "standard set" of animation authoring commands should look like. Therefore, in my study, I decided to show simple animated clips to participants, who were professional animators, and ask for a step-by-step procedure for animating the scene. For each step, the participants were then asked for multiple possible gestures to effectuate that particular operation. Clustering similar operations and gestures resulted in a set of common operations for VR-based animation, and a user-preferred gesture set for performing those operations.

MagicalHands: A Gestural Animation Tool in VR

Encouraged by the positive response of the professional participants, and motivated to put the study results to practice, I implemented a prototype tool for gesture-based animation. This tool, called MagicalHands, implements a subset of the gestural mappings and design guidelines inferred from the elicitation study. MagicalHands allows the creation of rigid motions and of particle systems using hand gestures, and implements the following key features.

– *Visual entities.* In addition to objects visible in the animation, functional meta-objects such as *particle emitters* and a *object shelf* to pick static objects from serve as explicit visual representations of affordances and boost immersion.

– *Extrinsic and intrinsic attributes.* Visual and mental clutter is managed by making *extrinsic* entities such as emission curve and spread only available when the relevant *particle emitter* object is selected. However, directly manipulated at-

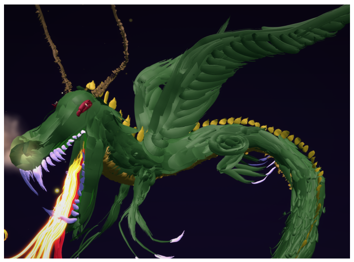


Figure 7: VR-sketches have free-form strokes, inconsistencies, and layered details. Utilizing such sketches for 3D animation could give rise to a new animation aesthetic. ©Jesse Weaver; used under CC-BY 3.0.

tributes such as the position and orientation of a 3D model can always be invoked by gesturally grabbing the object.
– *Creation process freedom.* Performance-based animation, keyframing, as well as physical simulation coexist in a coherent interface.

Users thoroughly enjoyed using the tool, and were able to successfully utilize both direct manipulation and particle systems to create animations. Fig. 6 shows an example.

Ongoing and Future Work

I'm currently exploring another creative application of immersive interactions—drawing and painting onto virtual objects. While there is no research in the immersive space that I know of, desktop-based tools for painting onto objects represented as triangle meshes have existed for decades [1, 8]. An inherent problem with desktop-based applications is that drawing is performed by projecting strokes onto objects by ray-casting, similar to SymbiosisSketch. Immersive realities allow users to orient a "paintbrush" arbitrarily in 3D and project virtual paint onto objects, a method utilized by recent immersive tools [17].

Unfortunately, painting accurately on geometrically complex objects still remains a gnarly task. Owing to the difficulty of positioning a paintbrush tool precisely on the virtual surface, strokes get projected far away from the intended spot, suffer from non-smoothness and discontinuities, and drawing long strokes is nearly impossible. I am conducting studies to fully characterize how and why obvious projection methods such as nearest-neighbour and ray-casting fail, and building a new method to project strokes onto triangle meshes. Other avenues of exploration are providing better visual cues to the user to help them draw accurately, and interactive guidance to the projection algorithm to create a more robust method that can adapt to diverse usecases.

Another application I plan to look at in the future is utilizing rough, incomplete sketches drawn using tools such as TiltBrush [13], Quill [18], and SymbiosisSketch [3] for animation. Character animation is especially attractive, since existing methods could be used to fit skeletal rigs to a sketched 3D character [5]. A recent algorithm [20] has successfully demonstrated the conversion of such sketches to watertight models for fabrication. Unfortunately, this method is not directly applicable to my application, since the characteristic "looseness" and incompleteness of the VR-drawn sketches is lost when converting to a watertight surface. I want to enable a novel aesthetic for animation by building an interactive tool which maintains the style conveyed by the artist, whether loose or watertight (Figure 7).

Conclusion

My dissertation focuses on the use of 3D interactions for immersive computer graphics applications. Immersive 3D creation is a burgeoning area and novel commercial tools [13, 18, 17] are already finding a variety of uses in the art and design communities. My experiments and prototype tools show a lot of promise for making immersive creation more intuitive, efficient, and joyful. In the future, I plan to continue investigating creative applications which can benefit from this novel input modality. I believe that the doctoral symposium will offer me a great opportunity to understand and expand the broader impact of my work.

Acknowledgments

I would like to thank my adviser Karan Singh for his immense support and guidance, and my long-term collaborators and mentors Rubaiat Habib Kazi and Tovi Grossman.

REFERENCES

[1] Allegorithmic | Adobe. 2019. Substance Painter. (2019). https://www.substance3d.com/products/substance-painter

[2] Rahul Arora, Ishan Darolia, Vinay P. Namboodiri, Karan Singh, and Adrien Bousseau. 2017. SketchSoup: Exploratory Ideation Using Design Sketches. *Computer Graphics Forum* (2017). http://dx.doi.org/10.1111/cgf.13081

[3] Rahul Arora, Rubaiat Habib Kazi, Tovi Grossman, George Fitzmaurice, and Karan Singh. 2018. SymbiosisSketch: Combining 2D & 3D Sketching for Designing Detailed 3D Objects in Situ. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (to appear) (CHI '18)*. ACM, New York, NY, USA. http://dx.doi.org/10.1145/3173574.3173759

[4] Rahul Arora, Rubaiat Habib Kazi, Danny M. Kaufman, Wilmot Li, and Karan Singh. 2019. MagicalHands: Mid-Air Hand Gestures for Animating in VR. In *Proceedings of the 32Nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. ACM, New York, NY, USA, 463–477. DOI: http://dx.doi.org/10.1145/3332165.3347942

[5] Ilya Baran and Jovan Popović. 2007. Automatic rigging and animation of 3D characters. In *ACM Transactions on Graphics (TOG)*, Vol. 26. ACM.

[6] Mark Billinghurst, Adrian Clark, and Gun Lee. 2015. A Survey of Augmented Reality. *Foundations and Trends® in Human–Computer Interaction* 8, 2-3 (2015), 73–272. DOI: http://dx.doi.org/10.1561/1100000049

[7] S.J. De Laet. 1994. *History of Humanity: Volume I*. UNESCO.

[8] David (grue) DeBry, Jonathan Gibbs, Devorah DeLeon Petty, and Nate Robins. 2002. Painting and Rendering Textures on Unparameterized Models. *ACM Trans. Graph.* 21, 3 (July 2002), 763–768. DOI: http://dx.doi.org/10.1145/566654.566649

[9] Mira Dontcheva, Gary Yngve, and Zoran Popović. 2003. Layered Acting for Character Animation. *ACM Trans. Graph.* 22, 3 (July 2003), 409–416. DOI: http://dx.doi.org/10.1145/882262.882285

[10] Tomás 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

[11] Koos Eissen and Roselien Steur. 2011. *Sketching: the basics ; the prequel to Sketching: drawing techniques for product designers*. BIS, Amsterdam. OCLC: 756275344.

[12] Facebook Newsroom. 2014. Facebook to Acquire Oculus. (2014). https://newsroom.fb.com/news/2014/03/facebook-to-acquire-oculus/

[13] Google. 2016. Tilt Brush by Google. (2016). https://www.tiltbrush.com/

[14] Tovi Grossman, Ravin Balakrishnan, and Karan Singh. 2003. An Interface for Creating and Manipulating Curves Using a High Degree-of-freedom Curve Input Device. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*. ACM, New York, NY, USA, 185–192. DOI: http://dx.doi.org/10.1145/642611.642645

[15] 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). http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4135646

[16] Noah Lockwood and Karan Singh. 2012. Finger Walking: Motion Editing with Contact-based Hand Performance. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '12)*. Eurographics Association, Goslar Germany, Germany, 43–52. http://dl.acm.org/citation.cfm?id=2422356.2422364

[17] Oculus. 2016a. Medium. (2016). https://www.oculus.com/medium/

[18] Oculus. 2016b. Quill. (2016). https://www.oculus.com/experiences/rift/1118609381580656

[19] Thammathip Piumsomboon, Adrian Clark, Mark Billinghurst, and Andy Cockburn. 2013. User-Defined Gestures for Augmented Reality. In *Human-Computer Interaction – INTERACT 2013*, Paula Kotzé, Gary Marsden, Gitte Lindgaard, Janet Wesson, and Marco Winckler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 282–299.

[20] Enrique Rosales, Jafet Rodriguez, and Alla Sheffer. 2019. SurfaceBrush: From Virtual Reality Drawings to Manifold Surfaces. *ACM Trans. Graph.* 38, 4, Article 96 (July 2019), 15 pages. DOI: http://dx.doi.org/10.1145/3306346.3322970

[21] Louis B. Rosenberg. 1993. Virtual fixtures: Perceptual tools for telerobotic manipulation. In *Proceedings of IEEE Virtual Reality Annual International Symposium*. 76–82. DOI: http://dx.doi.org/10.1109/VRAIS.1993.380795

[22] Ivan E Sutherland. 1968. A head-mounted three dimensional display. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*. ACM.

[23] Tvori. 2019. Tvori VR. http://tvori.co. (2019).

[24] Eva Wiese, Johann Habakuk Israel, Achim Meyer, and Sara Bongartz. 2010. Investigating the learnability of immersive free-hand sketching. In *Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium*. Eurographics Association.

[25] Jacob O. Wobbrock, Meredith Ringel Morris, and Andrew D. Wilson. 2009. User-defined Gestures for Surface Computing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1083–1092. DOI: http://dx.doi.org/10.1145/1518701.1518866