Physical Guides

An Analysis of 3D Sketching Performance on Physical Objects in Augmented Reality

Philipp Wacker
RWTH Aachen University
Aachen, Germany
wacker@cs.rwth-aachen.de

Adrian Wagner
RWTH Aachen University
Aachen, Germany
wagner@cs.rwth-aachen.de

Simon Voelker
RWTH Aachen University
Aachen, Germany
voelker@cs.rwth-aachen.de

Jan Borchers
RWTH Aachen University
Aachen, Germany
borchers@cs.rwth-aachen.de

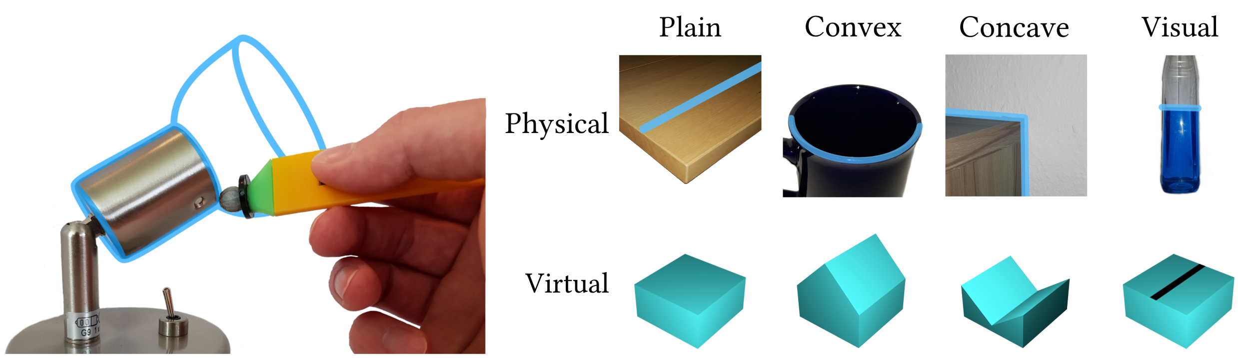

Figure 1: Augmented Reality lets users sketch directly on the surface of a physical object to design a new model that fits parts of the existing object (left). We classified the different surface guides that physical and virtual objects provide (right), and evaluated their effects on AR sketching performance.

ABSTRACT

Besides sketching in mid-air, Augmented Reality (AR) lets users sketch 3D designs directly attached to existing physical objects. These objects provide natural haptic feedback whenever the pen touches them, and, unlike in VR, there is no need to digitize the physical object first. Especially in Personal Fabrication, this lets non-professional designers quickly create simple 3D models that fit existing physical objects, such as a lampshade for a lamp socket. We categorize guidance types of real objects into flat, concave, and convex surfaces, edges, and surface markings. We studied how accurately these guides let users draw 3D shapes attached to physical vs. virtual objects in AR. Results show that tracing physical objects is 48% more accurate, and can be performed in a similar time compared to virtual objects. Guides on physical objects further improve accuracy especially in the vertical direction. Our findings provide initial metrics when designing AR sketching systems.

SUI '18, October 13–14, 2018, Berlin, Germany

ACM ISBN 978-1-4503-5708-1/18/10...$15.00
https://doi.org/10.1145/3267782.3267788

CCS CONCEPTS

• **Human-centered computing → Mixed / augmented reality**; **Empirical studies in HCI**;

KEYWORDS

Augmented reality; 3D sketching; tracing; motor ability; physical objects; guide classification

ACM Reference Format:

Philipp Wacker, Adrian Wagner, Simon Voelker, and Jan Borchers. 2018. Physical Guides: An Analysis of 3D Sketching Performance on Physical Objects in Augmented Reality. In *Symposium on Spatial User Interaction (SUI '18), October 13–14, 2018, Berlin, Germany.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3267782.3267788

1 INTRODUCTION

The recent advancement of Virtual Reality (VR) and Augmented Reality (AR) technology has rekindled an interest in using these techniques in 3D design tasks. Numerous research projects [1, 3, 16, 20, 31, 33] and products like Tiltbrush (www.tiltbrush.com) or Gravity Sketch VR (www.gravitysketch.com/vr) focus on sketching in mid-air to create 3D models in both VR and AR. Both approaches let users create and view models directly in 3D.

However, Augmented Reality is more closely related to the real, physical world [28] and enables sketching directly on existing physical objects, whether to extend them or to design a new, matching object. This is particularly helpful when designing an object that has to fit an existing object, such as a new lampshade to replace a broken one (Fig. 1, left), a handle for a paper cup, or a lid for a trash can. Being able to achieve this quickly and easily without extensive knowledge of professional modeling tools opens up 3D object design to novices. In VR, the user would first need to digitize the existing model. In AR, she can instead use the existing physical object directly, such as the lamp socket that the new lampshade should fit to. This also adds the benefit of haptic feedback from the physical objects' surfaces as guidance during the modeling task. For such a task, it is helpful to understand how well users can draw along a planned line on a physical object.

Sketching planar shapes in VR is more accurate when a flat physical surface is provided, compared to sketching in mid-air [2]. However, real objects, like a water bottle, have more complex surface shapes and offer a variety of guidance elements both visual (such as a printed line) and haptic (such as curves and edges). To explore this space of designing objects that can fit existing objects using AR, we first review related work from the sketching, 3D modeling, and haptics communities in AR and VR. We then provide a classification of the types of guidance that physical objects offer, and study their impact on 3D sketching precision in AR. Since AR also supports placing virtual objects into the real environment, like a desk model when planning a new office [17], our classification and study include virtual counterparts of each guidance type (Fig. 1, right). After discussing our findings, study limitations, and additional insights, we close with an outlook on future work.

In summary, this paper makes the following contributions:

- a classification of guidance types when drawing on existing objects in AR,
- the first lab study to quantify the impact of different guidance types on time and accuracy when sketching on non-planar physical and virtual models in AR.

Our classification may help others to structure experimental conditions and describe object features more consistently, while our study findings provide some initial metrics to researchers and designers of AR sketching systems to estimate the effects of these guidance types on drawing performance.

2 RELATED WORK

Research into making 3D modeling easier for novices is at the crossroads of four areas: *Sketching for 3D Modeling*, the creation of 3D models from 2D sketches; *Immersive Modeling* to create 3D geometry directly in VR and AR; *Haptics in VR*, adding haptic feedback to virtual environments; and using *Physical Objects as Guides* that help when creating digital models.

2.1 Sketching for 3D Modeling

Particularly in the early design stages, many professionals prefer sketching their ideas in 2D on paper. Several research projects have aimed to bring the simplicity and ease-of-use of this process to 3D modeling. Olsen et al. [30] provide a survey of techniques and challenges. In particular, they emphasize the very divergent motivations and styles of sketching, and resulting technical needs, of different user bases, from artists to engineers.

Many approaches interpret 2D sketches on a tablet to create new 3D geometry. They derive the missing depth information from sketches from different viewpoints [3, 16], strokes positioned on different drawing planes [12], or graphical cues in the 2D sketch [43]. Others use a two-step approach: after creating a scaffolding of straight lines, subsequent curved strokes are interpreted in relation to that scaffolding [36]. More generally, strokes can be interpreted to 'wrap' around existing 3D models, e.g., of a mannequin [31, 35].

Other techniques use context provided by related objects to determine the 3D characteristics of a 2D drawing. Lau et al. [24] interpret lines on a photo in the context of other objects in the image, such as cupboards or tables, to infer a 3D model fitting into that scene. In Napkin Sketch [42], after the user specifies a drawing plane on her tablet in relation to a "napkin" displayed in AR in front of her, the system maps her 2D strokes on the tablet directly onto that plane and visualizes the resulting model in AR, on top of the live camera image.

To improve the quality of the drawing created, researchers have proposed a variety of guidance systems. PapARt [25] lets users manipulate a 2D projection of a 3D scene, then simplifies it for easy tracing by the user. Similarly, Flagg et al. [9] decompose an artwork into layers that are projected onto a canvas sequentially to help the user replicate it. Other systems help aligning strokes by displaying visual guides [15] or adjusting the stroke afterwards [8]. Rivers et al. [34] project guides onto a solid raw sculpture to indicate where to add or remove material to reach a predefined shape.

Based on studies of how well humans can follow paths [32], Cao and Zhai [6] provide a model of human pen strokes that reveals that, while humans slow down at stroke corners, the exact angle of the corner hardly impacts speed.

2.2 Immersive Modeling

This research area looks at interaction techniques to directly create 3D models in mid-air. This requires tracking input devices in space and aligning the visualization to create the impression of creating strokes in mid-air, as in the FreeDrawer system [40], for example. Jackson and Keefe [20] let users arrange digital scans of their 2D sketches in a virtual 3D environment, then create a wireframe model by selecting and "lifting" individual strokes from those sketches, and pull out surfaces between the strokes. WireDraw [44] supports creating a physical wireframe of a known 3D model with a 3D extruder pen, by displaying stroke guides in AR over the emerging physical object. In order to allow more precise printing of wireframes, Peng et al. [33] developed a system that allowed the user to create and manipulate 3D geometry in AR while a robot simultaneously prints the physical model in the same space.

Specifying curves is a recurring challenge in mid-air modeling. Proposed methods range from movement-sensing input strips [4] to two-handed tape-drawing gestures [13, 22].

Fleisch et al. [10] use both tape-drawing and other input devices for sketching on a semi-immersive virtual table, with physical 'input planes' to align a digital plane with the real table surface as an aid in drawing strokes. Mockup Builder [7] combines the precision of using a touchscreen with the realistic viewpoint manipulation of a VR headset for gesture-based modeling.

Similarly, Arora et al. [1] also combined immersive sketching with a physical drawing surface. Users can sketch in Augmented Reality as well as define drawing planes. Subsequent sketches on the physical tablet are then projected onto the drawing plane. The users benefitted from the expressiveness of free mid-air sketching while also having the precision of a physical drawing surface available.

While performance of freehand sketches improves with practice [41] and users can recreate patterns and positions without visual feedback [14], the missing haptic feedback still affects their precision. For VR, Arora et al. [2] showed that displaying visual guidelines, such as a surface and/or optimal path, already improves the drawing accuracy (overall and projected deviation from the optimal stroke) of freehand strokes, while providing a physical surface as guide improves it dramatically. The physical surface used in their study was a flat board and the target circle was aligned to this board. The study presented in this paper also measures the stroke accuracy under different guidance conditions. Compared to Arora et al. [2], the surfaces in this study were not flat surfaces and participants had to draw around physical objects in AR.

2.3 Haptics in VR

Missing haptic feedback is a well-known major limitation of VR [5, 27], not only for gesture accuracy as discussed above [2], but also immersion [18]. Even passive haptic elements already improve immersion in VR, e.g., a ridge on the floor improves the feeling of standing at the edge of a cliff [18].

Active haptic feedback increases both immersion and accuracy of input in VR: Creating haptic constraints improves the capability to draw 3D curves [22], and stimulating arm muscles electrically can simulate the haptics of virtual objects [26].

Finally, static physical objects can assist input in VR. Performing gestures in relation to physical printouts can simplify input to a VR system [19], even though this study did not provide in-place visual feedback of the stroke. 3D printouts of corals, for example, have been used to navigate data about them in VR [23], and physical maze elements have assisted novices creating VR mazes [11].

2.4 Physical Objects as Guides in Personal Fabrication

Personal Fabrication often requires aligning virtual models and physical objects, for example when designing an object to 3D-print that should fit around or inside an existing object.

Zhu et al. [45] use physical objects such as pens during the 3D printing process to create exact cutouts on printed objects. MixFab [39] lets users place small physical objects behind a see-through display and create virtual models aligned to them, e.g., to cut holes in the virtual object that fit the physical object. Weichel et al. [38] present physical measurement tools, such as a caliper, that communicate with a digital development tool to send physical measurements to the computer or digital values back to the physical tool. In ModelCraft [37], users sketch directly on folded paper objects covered with a printed marker pattern that lets a special pen send each sketched edit and annotation back to the linked digital model.

In summary, while research has been exploring sketching in AR from several directions, there has been no quantitative study of how different types of physical and virtual guidance affect the precision we can expect when users sketch in AR.

3 CLASSIFICATION OF GUIDANCE TYPES

To structure our study, we first introduce a classification of guidance types that objects may offer to a user tracing shapes on their surface with a pen to create strokes that are aligned to the object. While many factors affect drawing on physical objects, from material properties such as hard, soft, rough or smooth surfaces, to object size, location, and orientation, we focus on local object features that may guide a stroke.

In the simplest guided case, the user is drawing on a flat surface, such as a table (Fig. 1, top right). If we consider this surface as the xy-plane, then this restricts movement in the $-z$ direction. This is a hard constraint since the user cannot press into the table surface. It reduces pen movement by half a degree of freedom. To avoid lifting the pen off the surface in the $+z$ direction, the user exerts some pressure while tracing. This removes another half degree of freedom. However, this is only a soft constraint, since the user can still move in that direction. Therefore, the guidance to draw on a flat surface can be seen as an even mix of hard and soft constraints.

Nonplanar surfaces provide additional tracing guides to the user, such as the edge of a table. Such guides aim to reduce the movement to a line, and enable the user to trace particular features of the object. While none of these nonplanar surface features completely remove another degree of freedom (this would require locking the pen tip into a surface rail), how hard or soft their constraint is depends on the physical shape of the guide. A *concave* form along the line the user wants to trace, e.g., around the neck of a vase, is a harder constraint since the user’s pen pressure will push the pen tip towards the guide. A surface that is *convex* along the line the user is tracing, such as the opening of a mug, is a softer constraint since the user can easily slip off the ideal line. The more concave or convex the surface, the stronger the constraint. The extreme cases of these guides are especially common and worth studying: concave edges, as when tracing the inside edges of a box, and convex edges, like the outside edges of a box, or the table edge mentioned above.

Of course, visual markings on a surface can also guide the user in drawing a particular line. They can be natural, like the grain in a wooden table or the water line inside a bottle, or artificial, like a printed line on a book cover. We refer to these guides as *visual* guides. They are soft constraints, because they remove a degree of freedom when tracing them, but without providing any physical, haptic guidance.

Therefore, tracing a particular feature of a physical object can be understood as a limitation of two degrees of freedom involving both hard, physical and soft, “logical” constraints [29]. The first limited degree constrains movement to the surface of the object, as with a free-hand stroke drawn on a table. At this stage of the classification, there is no difference between drawing on a flat surface like a table or around a cylindrical object like a bottle. In both cases sketching is limited to the surface by equal measures of hard (cannot press into the object) and soft (should not lift pen off the surface) constraints. A second limited degree of freedom constrains movement to along

Constraint	*Virtual*	*Physical*
No guide	Virtual table surface	Real table surface
Visual	Pen stroke on a virtual sketch	Waterline in a bottle
Concave	Inside of a virtual bucket	Intersection of shelf & wall
Convex	Edge of a virtual desk	Opening of a wine glass

Table 1: Examples for all combinations of initial constraint & subsequent limitation to a line.

a one-dimensional line, straight or curved, on the object, using its physical shape or surface markings.

So far, we have only considered drawing on physical objects in AR. However, since AR supports showing virtual objects in the real world, tracing them should also be considered for comparison. For example, when planning a kitchen in AR, a user may want to trace on a virtually displayed working surface to outline the cutout for the sink. In this case, the first limited degree of freedom that constrains movement to a surface is already a soft constraint, because it is physically possible to penetrate the surface of a virtual object. Similarly, convex and concave surface features on virtual objects are only soft constraints, as are surface markings.

In conclusion, the guidance types on an object can be seen as limiting the degrees of freedom for sketching. The first limitation guides the free-hand movement to a surface by constraining one degree of freedom (*surface guidance*). This guidance can be either *physical* or *virtual*. On physical objects, this constraint is hard for one half degree of freedom and soft for the other half. On digital objects, it is entirely a soft constraint. Movement can be limited further to tracing a one-dimensional line or curve by *visual* guides and object shape (*line guidance*). *Concave* shapes provide more guidance, reducing the degree of freedom more than *convex* shapes, and curvature also increases guidance, with *concave* and *convex* edges as extremes. This results in 8 combinations (Table 1).

4 TRACING STUDY

Our study was designed to quantify the effect of different guidance types on tracing time and accuracy. The results can help AR sketching systems to filter raw pen movements based on the context to determine intended strokes. Knowing when users want to move to mid-air and not follow a certain guidance on the object anymore, could be used to switch between smoothing the performed stroke to "snap to the real object" and recording the freeform mid-air sketch.

To focus on this influence, we chose the basic task of drawing on the surfaces of fixed, upright objects and kept other properties such as size consistent. The size of our objects matched things typically created in affordable 3D printers' small build volumes. We chose a cylinder and a cuboid as our basic object types, around which the user had to draw a *circle* and *square* respectively. This way we covered both continuous and non-continuous motions. We decided to use fixed stationary objects to avoid effects of varying grips and of rotating the object instead of moving the pen to draw. It also let us include *virtual* objects without adding confounding variables such as handling and moving of digital objects. We measured accuracy by comparing the trace to the optimal shape.

4.1 Experimental Design

For each object (cylinder and cuboid), we included both *physical* and *virtual* models as conditions to reflect the *surface guidance* described before. Our guidance types led to four conditions: *no guide* as baseline, *visual*, *concave*, and *convex*. For this study, we focused on edges as the most pronounced and common *convex* and *concave* surface features.

The result is a 2×2×4 design: *shape* (*circle* / *square*) × *surface guidance* (*physical* / *virtual*) × *line guidance* (*no guide* / *visual* / *concave* / *convex*).

The sequence of independent variables was counterbalanced with a Latin square. Participants performed 80 strokes total, five strokes in each condition. They could practice each condition before their five strokes. A session took 45–60 min.

4.2 Participants

Two of our initial 16 participants expressed general issues with detecting the position of objects displayed by our AR headset. We excluded their results from the evaluation, and recruited two replacements to fill the Latin square again. Of our final 16 participants (4 female, 19–29 years, Mean: 24.9 years, SD: 2.4 years, all able-bodied), 9 wore glasses, but were near-sighted at a level sufficiently low to use our headset without glasses. 11 had no prior experience with AR in general or VR drawing tools; 4 had experience with AR; 1 had no AR experience, but had used a VR drawing tool in the past; no one had experience with both.

4.3 Apparatus

Our 3D-printed physical objects for tracing were 16 cm high, 8 cm wide, and could be mounted normally or upside down. A different *line guidance* was placed at 4 cm from each end. This let us combine the *no guide* and *visual* conditions on each green object, and the *convex* and *concave* conditions on each red object (Fig. 2, back). The object was mounted on a plate in front of the participant. The plate was attached to the table to avoid accidental movements.

4.3.1 Pen Design & Tracking. We created a custom pen (Fig. 2, front) tracked by a VICON motion tracking system that used 6 high-speed infrared cameras to track reflective markers from different angles at 100 fps with sub-millimeter accuracy. Similar to other projects (e.g., [20]), our pen featured markers at the end and tip. Using a spherical marker as the tip both improved tip tracking stability and prevented user confusion when the alignment of the virtual line rendering drifted slightly around the center of the physical tip. The pen included two buttons for inking and calibration, and a Bluetooth LE module to send their states to a receiver. A Mid-2012 MacBook Pro running our user study software extracted the pen tip position in 3D space from the VICON data. While the Inking button was pressed, the software recorded the pen tip position as a path, and forwarded it to a Microsoft HoloLens headset to render the path into the user's view.

4.3.2 Visualization (Registration & Rendering). Using a Microsoft HoloLens headset allowed us to visualize the drawn stroke and the *virtual* objects with high stability. This required aligning the VICON and HoloLens coordinate systems. Our calibration setup used the VICON calibration wand and a visual marker that the HoloLens could track. Knowing their positions in reality, we could roughly align both coordinate systems. To address limitations of the visual

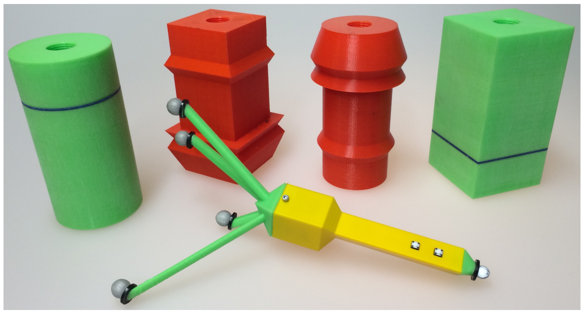

Figure 2: Front: Custom pen used in the study. Back: Physical objects used in the study. For each *shape* condition, we had 2 physical objects to reflect the *no guide* and *visual* condition as well as the *convex* and *concave* condition.

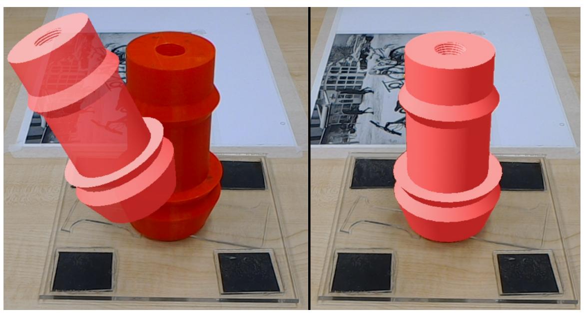

Figure 3: After detecting the visual marker, VICON and HoloLens coordinate systems are not perfectly aligned yet (left, exaggerated). We manually adjust them so that the virtual model is aligned precisely with the physical model (right).

tracking of the HoloLens, we then fine-aligned the coordinate systems manually by adjusting rotation and location of a digital model to fit a real-world counterpart placed at a known point (Fig. 3). This allowed us to calculate where a point measured by the VICON would need to be displayed in the HoloLens, to render a sphere on top of the physical pen tip as it moved around the room, and to render strokes in place while inking (Fig. 4). While the user pressed the Inking button, we rendered a line with 30 fps using the most recent points forwarded to the HoloLens, to visualize the path the user was drawing. We occluded those parts of the path behind the model in both the physical and the virtual condition to preserve realism, based on pilot tests (Fig. 4 top, left).

4.4 Study Procedure

Participants sat at a table inside the VICON's tracking volume (Fig. 4, top, right). They were allowed to move their head and torso, but were asked to remain seated. Each trial started by showing the object to trace around. In *physical* conditions, we screwed the object onto the mounting plate, and asked participants to grab the object with their non-dominant hand while drawing. In *virtual* conditions, we asked participants to rest their non-dominant hand on the mounting plate.

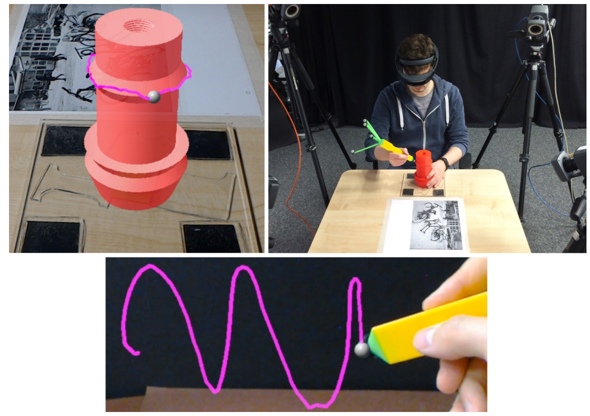

Figure 4: After aligning the coordinate systems, points reported by the VICON can be rendered into the viewing area of the HoloLens (bottom). In our study, participants were asked to draw a stroke around virtual objects (top, left) and physical objects (top, right).

We asked our participants to draw around the object with a regular drawing speed, while keeping precision in mind. In the *virtual* conditions, the participant was allowed to move their hand and the pen through the virtual object.

During the implementation of the system, we observed that the HoloLens occasionally adjusts its coordinate system due to updated tracking information from the environment. If this happened after calibration of our system, the coordinate systems of VICON and HoloLens became misaligned, making the real and rendered pen tip deviate from each other by up to 10 mm. We asked participants to mention any offset to us during the study and also inquired about the correct alignment occasionally throughout the session. In case of a misalignment, we re-synchronized the coordinate systems before continuing. This calibration was necessary once each participant first mounted the HoloLens, because a user's individual physiology affects the alignment of the HoloLens. We encountered the aforementioned need to recalibrate during sessions only twice, and re-ran the last trial after recalibration.

4.5 Measurements

Similar to the study from Arora et al. [2], we processed the data for each stroke with a low-pass filter averaging over a 10 frame window, created a path from the resulting points, and then resampled the path to 100 equidistant points for evaluation. This removed a potential bias due to the higher point count in areas such as corners where participants slowed down. We compared these resampled points to the optimal stroke to calculate 5 measurements:

4.5.1 Mean Deviation in 3D. We calculated the shortest distance from the target shape for each point and calculated the mean. This represents the mean deviation from the target shape [2].

4.5.2 Mean Deviation in x & y Direction. We projected the sampled points onto the surface plane of the table, and computed the mean difference from the projected target shape. This measured the

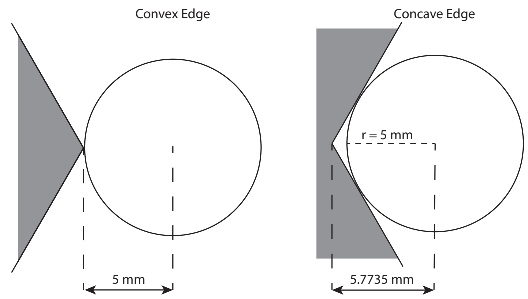

Figure 5: Optimizations made to the target stroke for the *physical* conditions to correct the inability to draw the perfect stroke.

effect of *surface guidance* (*physical* or *virtual*). In the *physical* conditions, this corresponded to 'lift-offs'. The calculation corresponds to the *Mean Projected Deviation* from Arora et al. [2].

4.5.3 Deviations in z Direction. For this, we evaluated only the z coordinate of each sampled point and calculated the deviation from the target height. For the *no guide* condition, we set the height of the target stroke to the height of the first recorded drawing position for each stroke. This deviation evaluates the effect of *line guidance*, as it measures the deviation in the dimension unconstrained by *surface guidance*. We split this value up into two sub-classes:

(1) The *mean absolute deviation* from the target height.
(2) The *mean directed deviation* from the target height. This allows us to evaluate whether a stroke was mainly above or below the target height.

Since the line drawn originated from the center of the spherical marker at the pen tip, the marker displaced the user's input on physical surfaces by its radius of 5 mm, making it impossible to perfectly trace physical surfaces. To account for this, we virtually enlarged all physical target shapes by that radius for our calculations. The *physical*, *concave* condition required another enlargement to a total of 5.77 mm since the radius of the pen tip marker was larger than the concave opening, keeping it further away from the concave edge (Fig. 5). Adjustments were not necessary for the *virtual* conditions since the user could penetrate object surfaces to align strokes.

4.5.4 Stroke Duration. We also recorded drawing time by measuring the duration between the first and last inking operation for each stroke.

4.6 Results

Due to measurement issues, we discarded 20 strokes of the 1280 recorded. For every participant, we averaged the dependent variables over the 5 trials. To analyze the effect on our experimental conditions, we performed mixed-effect ANOVAs with the user as a random variable on the log-transformed measurements. All post-hoc pairwise comparisons were performed using Tukey HSD tests.

4.6.1 Mean Deviation in 3D. Both guidance conditions (*surface guidance*: $F_{1,225} = 263.33, p < .001$; *line guidance*: $F_{3,225} = 6.17, p < .001$) as well as their interaction ($F_{3,225} = 6.00, p < .001$) had a significant effect on the mean deviation in 3D. The means and results of the post-hoc tests can be seen in Table 2.

Condition	Significance			Mean	Standard Deviation
Physical	A			4.94	1.47
Virtual		B		9.45	3.73
Convex	A			6.69	3.32
Visual	A			6.89	3.34
Concave	A			7.11	4.12
No guide		B		8.09	3.58
Physical, concave	A			4.12	1.24
Physical, visual	A	B		4.77	1.08
Physical, convex	A	B		4.96	1.15
Physical, no guide		B		5.92	1.77
Virtual, convex			C	8.43	3.84
Virtual, visual			C	9.01	3.50
Virtual, concave			C	10.11	3.79
Virtual, no guide			C	10.26	3.64

Table 2: Means and standard deviations of *mean deviation in 3D* for main effects and interactions (in mm, rounded to two decimal places). Rows represent conditions, horizontal lines separate main effects and interactions. Rows not connected by the same letter are significantly different.

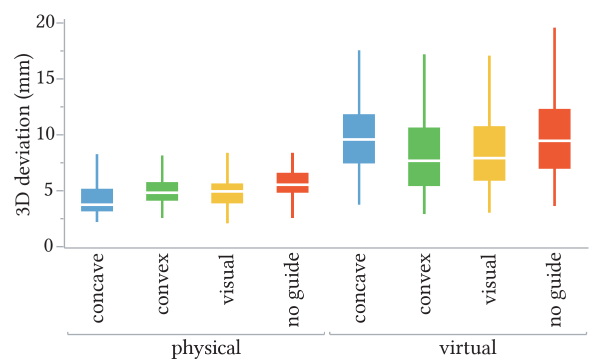

Figure 6: Interaction effect of *surface guidance×line guidance* on *mean deviation in 3D*. The *physical* conditions have the least deviation (significant). The *physical, concave* condition is also significantly different to the *physical, no guide* condition. Whiskers denote 1.5×IQR.

4.6.2 Mean Deviation in x&y Direction. For mean deviation in x&y direction, *surface guidance* ($F_{1,225} = 333.05, p < .001$) and the interaction *surface guidance×line guidance* ($F_{3,225} = 5.07, p < .01$) showed significant differences. Comparing the means shows that *physical* (M: 2.81 mm, SD: 1.17 mm) deviated less than *virtual* (M: 6.92 mm, SD: 2.98 mm). Post-hoc tests of the interaction also show that the *physical* conditions deviated significantly less than the *virtual* conditions.

4.6.3 Deviation in z Direction. The overall deviation in z direction showed a significant effect of *surface guidance* ($F_{1,225} = 82.64, p < .001$), *line guidance* ($F_{3,225} = 6.21, p < .001$), and the interaction between them ($F_{3,225} = 7.90, p < .001$). See Table 3.

Comparing the directed deviation in z direction shows significant differences due to *surface guidance* ($F_{1,225} = 100.23, p < .001$). Also the interaction between *surface guidance* and *line guidance*

Condition	Significance		Mean	Standard Deviation
Physical	A		3.44	1.21
Virtual		B	5.22	2.44
Convex	A		4.11	2.06
Concave	A		4.22	2.80
Visual	A		4.24	1.89
No guide		B	4.75	1.49
Physical, concave	A		2.78	1.07
Physical, convex	A		3.20	0.84
Physical, visual	A		3.28	0.96
Physical, no guide		B	4.49	1.23
Virtual, no guide		B	5.01	2.50
Virtual, convex		B	5.01	1.70
Virtual, visual		B	5.20	2.11
Virtual, concave		B	5.65	3.24

Table 3: Means and standard deviations of *mean deviation in z direction* for main effects and interactions (in mm; rounded to two decimal places). Rows not connected by the same letter are significantly different.

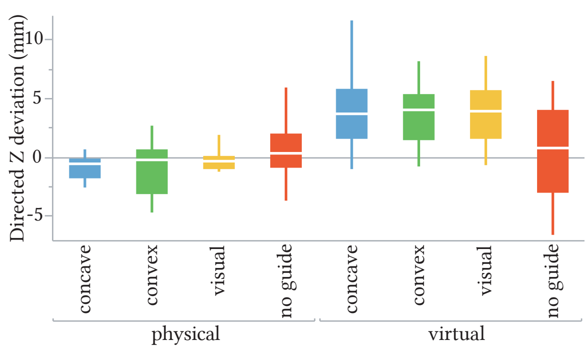

Figure 7: Interaction effect of *surface guidance*×*line guidance* on *mean directed z deviation*. The *physical* conditions except *no guide* were mainly drawn beneath the target line. The *virtual* conditions with a guide were drawn significantly above the target line. Whiskers denote 1.5×IQR.

is significant ($F_{3,255} = 13.25, p < .001$). The means show that users drew slightly beneath the target line on *physical* objects (M: -0.62 mm, SD: 2.03 mm) compared to *virtual* objects (M: 2.88 mm, SD: 3.81 mm). Analyzing the interaction further shows that especially the strokes on *virtual* objects with any *line guidance* significantly deviate in the positive *z* direction (Fig. 7).

4.6.4 Stroke Duration. Regarding the duration, *shape* ($F_{1,225} = 41.29, p < .001$) shows significant differences. Participants drew significantly faster around the cylinder (M: 7.85 s, SD: 3.68 s) compared to the cuboid (M: 9.95 s, SD: 5.04 s). There was no significant interaction effect with *surface guidance*×*line guidance*. However, the graph of the interaction indicates a trend that the *physical, concave* condition could be faster to draw around (Fig. 8).

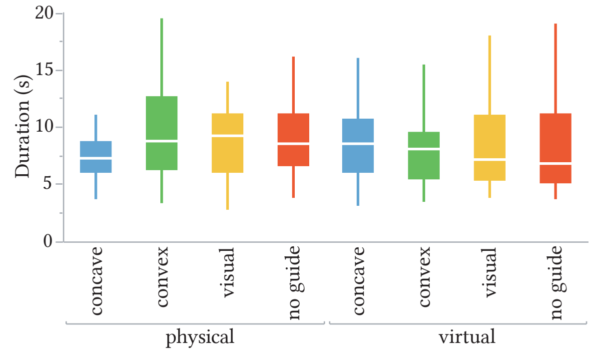

Figure 8: Interaction effect of *surface guidance*×*line guidance* on the *mean duration* of a stroke. While the duration for each condition is similar, there is a trend that sketches on the *physical, concave* object could be faster. Whiskers denote 1.5×IQR.

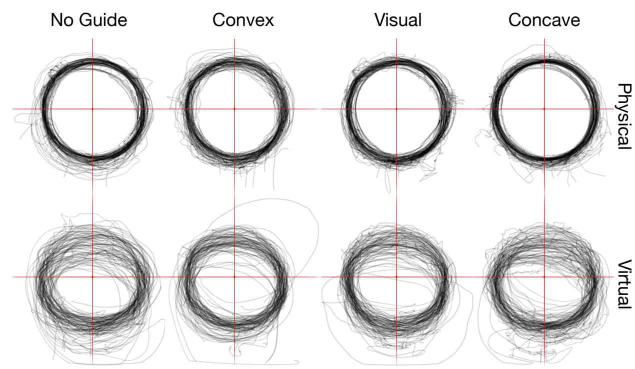

Figure 9: Top rendering of the interaction *surface guidance*×*line guidance*. The bottom half of each condition represents the front view. *Physical* strokes show less deviation compared to *virtual* strokes.

4.7 Front-to-Back Comparison

Studying aggregate renderings of the strokes performed, as in Fig. 9, we noticed a pattern that motivated us to compare accuracy on the front- and back-facing halves of each object, to evaluate how well users could continue strokes they could no longer see. We split recorded points into a front-facing and a back-facing half. We computed the mean 3D deviation, mean *x&y* deviation, and mean absolute *z* deviation for both halves. We re-did the evaluation and report all results related to the new variable *side* as well as all interactions involving it.

For mean 3D deviation, we found a significant effect of *side* ($F_{1,465} = 3.93, p < .05$). Comparing the means shows that *front* (M: 7.05 mm, SD: 4.04 mm) deviated less than *back* (M: 7.29 mm, SD: 3.60 mm). However, there were no significant interaction effects.

For mean *x&y* deviation, there was a significant interaction effect *shape*×*surface guidance*×*side* ($F_{1,465} = 5.87, p < .05$). However, the post-hoc tests show only significant differences between *physical* and *virtual* conditions regardless of *shape* or *side*. Since the stroke

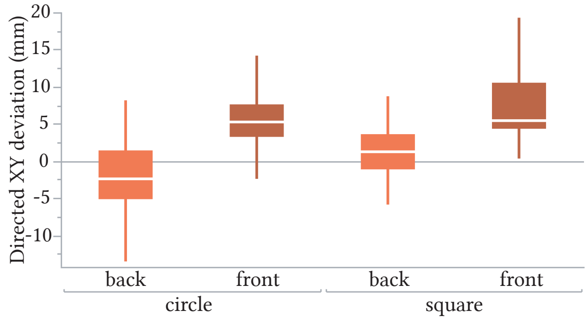

Figure 10: *Directed x&y deviation* for *virtual* objects separated by *side* and *shape*. On the *front* of the object, the deviation is mostly on the outside. On the *back* the strokes also deviate inside the object more frequently.

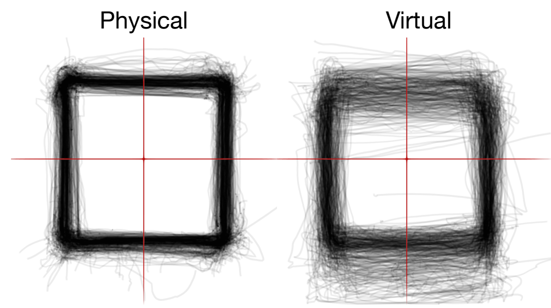

Figure 11: Top rendering of the accumulated strokes for the *square* separated by *surface guidance*. The bottom half of each condition represents the front view. *Virtual* strokes appear slightly rotated counter-clockwise and more spread out at the front and back than the sides.

renderings indicate that strokes on the *virtual* objects deviate more into the object on the *back* while deviating out on the *front*, we also calculated the directed *x&y* deviation—points inside the object were given a negative value, points outside a positive value. While the overall means show similar deviation (*virtual*, *front*: M: 7.18 mm, SD: 3.89 mm; *virtual*, *back*: M: 6.38 mm, SD: 2.79 mm), the directed means show differences (*virtual*, *front*: M: 6.35 mm, SD: 4.11 mm; *virtual*, *back*: M: -0.65 mm, SD: 5.02 mm). Fig. 10 shows this for both *shape* conditions.

Analyzing the mean absolute *z* deviation showed a significant main effect for *side* ($F_{1,465} = 19.92, p < .001$) and an interaction effect for *surface guidance*×*line guidance*×*side* ($F_{3,465} = 4.55, p < .01$). Comparing the means shows that *front* (M: 3.98 mm, SD: 2.07 mm) deviated less than *back* (M: 4.75 mm, SD: 2.62 mm). Post-hoc tests for the interaction echo similar trends that *surface guidance* is responsible for an increased precision and has a stronger influence than *side*.

4.7.1 Qualitative Observations. Another intriguing observation was that the *virtual square* shapes showed a slight counter-clockwise rotation (Fig. 11). Their sides were also traced more accurately than their front and back. Both observations appeared in all *virtual*, but no *physical* conditions.

5 DISCUSSION

Our results indicate that both *surface guidance* and *line guidance* affect the performance of drawing on objects. In particular, *physical* objects improve drawing accuracy in all metrics measured. Strokes on *physical* objects deviate less from the target, both overall and in each direction. This shows that the hard constraint of a surface supports the user more than its soft “lift-off” constraint.

Evaluating the effect of *line guidance* shows that having *no guide* to follow reduces stroke accuracy. This highlights that any guide at all helps the user to continuously correct deviations from a target line. Especially for *physical* objects, it is an interesting finding that a *visual* guidance performs similar to the guidance given by a physical edge. This means that sketching tools should not only pay attention to the geometry of an object but also its texture.

The interaction of *surface guidance* and *line guidance* is showing interesting results. On *physical* objects any guide greatly improves the precision, especially to keep the intended height, while on *virtual* objects, the deviation in height was similar between all *line guidance* conditions. However, that there are no significant differences between the different guidance types means that the increased hard constraint by the *physical*, *concave* condition did not have as much impact as we expected. Even though the measured deviation for the *physical*, *concave* condition were the lowest on all measurements, the difference to the others was not significant.

Another interesting finding is that participants performed their stroke on average below their target line on *physical* objects while drawing higher on *virtual* objects. The only exception is the *no guide* condition. As we adjusted the target height to the initial inking height for the *no guide* conditions, this leads to the assumption that participants varied around their initial height but frequently misjudged this height in the *virtual* guide conditions. One possible explanation for this is the issue of detecting depth in virtual environments [21]. As participants were seated at the table and were looking down at the objects on the table, a misjudgment in depth could lead to a higher performed stroke than intended.

The only effect that *shape* had on the drawing performance is that it took participants longer to draw a *square* than a *circle*. This may be due to the non-continuous corners of the stroke, which require more attention. The abrupt change in direction likely forced participants to slow down. This is similar to the findings from Cao and Zhai [6], and Pastel [32].

While our results show differences in the mm range, these already matter for certain modeling tasks, and previous studies indicate that such differences are likely to increase with the size of the target shape [2].

After our main evaluation, we also compared the performance on the *front* and *back* of the object, since the stroke visualizations indicated differences. We found that participants generally deviated more on the *back* of the object. This is most likely due to the missing visual indication of where to trace, so that the user has to rely on physical guides and motor memory to continue the trace. Interestingly, there were no significant interaction effects regarding *surface guidance* or *line guidance* for overall or *x&y* deviation. Looking at the top-view renderings suggests that participants deviated more into the object on the back of the object, while drawing outside the object in front. However, looking at the means showed that while many strokes were performed inside the object on the *back*, the directed deviation is close to zero suggesting that the deviation outside of the shape is similar. This is also apparent in the high standard deviation for the directed deviation. Since the overall deviation on the *front* is similar to the directed deviation, this means that most strokes were drawn outside of the object. As strokes inside the object were occluded, this means that participants were able to detect whether their stroke was inside the object but had problems determining how far they were away from the surface. Further studies are necessary to fully explore the effect of guides under different visibility conditions, and how visualizing both path and guide on the back of objects affects tracing performance.

The qualitative observations about a higher accuracy on the sides of a *square* in the *virtual* condition might be explained by the issues humans have with detecting depth in virtual environments [21]. The sides present a clearer edge in *x&y* direction, so that participants could judge the correct position more easily. The front and back do not present such a clear border. This can also explain the 'rotation' of squares in the *virtual* condition. As participants follow their stroke on one side, they had to judge whether they were still 'on' the side and continue drawing. The motion that followed was the 'blind' stroke along the back of the object. As participants show the tendency to move into the object, this means that, coupled with overshoots when detecting the edge on the left side, they draw a slightly rotated square. This assumption only explains the offset for counter-clockwise strokes (75.6% of strokes performed). An intriguing question for future studies is whether the same phenomenon occurs when drawing in clockwise direction. Participants' handedness is also a likely factor in this. Further analysis of the influences of drawing direction and handedness on sketching offsets on *virtual* models thus appear to be an interesting research direction.

The pen used in our study had a 10 mm diameter spherical tip similar to the pen in the study by Jackson and Keefe [20], and the stroke drawn originated from the center of this sphere. As explained in the Experimental Design, this meant that it was not possible to draw a stroke visibly on the physical surface. While we applied corrections to the collected data in order to alleviate that effect, further analysis is needed to see whether the performance changes when using a finer tip. Finally, we did not look at the orientation of the pen in this study. Especially with a large, spherical tip, the mental model of where on that sphere the user considers the "drawing point" to be for different surface–pen angles is an intriguing direction for further research.

5.1 95% Neighborhood

Based on the results of our study and the identification of relevant differences, we computed thresholds for various conditions that cover 95% of the recorded points.

As *surface guidance* had the most influence on drawing performance, we computed the 95% neighborhood of both *physical* and *virtual* conditions. In the *virtual* conditions the threshold is twice as large as in *physical* conditions (*virtual*: 22.87 mm; *physical*: 11.15 mm). For *virtual* objects, the difference between *front* and *back* (*front*: 20.34 mm; *back*: 25.05 mm) is greater than for *physical* objects, for which it is practically negligible (*front*: 11.35 mm; *back*: 10.93 mm).

Since *line guidance* had an effect particularly for *physical* objects in the way that any guide at all improved the accuracy, we computed thresholds for those conditions. On *physical* objects with *no guide*, 95% of the recorded points fall within 14.13 mm of the target stroke while any guide (*visual*, *convex*, *concave*) improves the size of this area to 9.78 mm around the target line.

These thresholds could be used by sketching systems to decide whether a user is still intending to follow a particular line on an object (that the system could then 'snap' the line to) or whether she is intentionally deviating to move to sketching in mid-air.

6 CONCLUSION & FUTURE WORK

Immersive modeling is a promising approach to make generating simple 3D models more accessible to nonprofessionals. Modeling in AR has the added benefit over Virtual Reality of visual and haptic real-world feedback. This is especially beneficial in the area of Personal Fabrication, when designing models that should attach to or align with existing objects.

We classified the types of guides that existing objects offer for drawing on them. All objects provide *surface guidance* when drawing on them. On *physical* objects, users cannot push into the surface (hard constraint) and should not lift-off the surface (soft constraint). Surface structures like *convex* or *concave* edges guide the user further in drawing along a particular line or curve on the object (*line guidance*). Such lines can also be merely *visual* (e.g., printed). Since AR also allows placing virtual objects into the real world, all guidance types can also exist on *virtual* objects.

In our lab study, we quantified the effects these different guidance types have on both accuracy and time needed to complete a stroke when drawing around an object. For this, we synchronized a VICON motion tracking system with a Microsoft HoloLens, and measured how far participants deviated from the optimal stroke for each guidance type. We found that the deviation was the lowest for the *physical* conditions.

In an additional analysis, we found that participants deviated more outside of the *virtual* object in the *x&y* dimension when the target stroke was on the *front* of the object while being more evenly distributed inside and outside on the *back* of the object.

Our study covers only a small area in the design space of drawing interactions with physical and digital objects in Augmented Reality. We focused on a single-handed task of drawing around a static object. Holding the object to trace in the other hand creates interesting questions regarding the bi-manual tracing interactions. These questions include qualitative directions such as different strategies of performing the stroke as well as performance related questions such as the effect on accuracy when performing a bimanual stroke. Further studies should look at the effect of different object orientations, sizes, surface structures, and materials, as well as different pen styles, on drawing performance. AR can also show things that are impossible to see in reality. For example, we occluded the stroke when drawing behind an object, since pilot tests found other approaches to be too confusing, but the stroke could still be made visible, e.g., as a dotted line, with potential benefits to the user. Investigating these options and how they can be used to improve modeling tasks in Augmented Reality are promising directions to take in this field.

ACKNOWLEDGMENTS

We want to thank all our participants for taking part in the study. This work was funded by the German Federal Ministry of Education and Research as part of the Photonics Research Germany (13N14065).

REFERENCES

[1] 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 *Proc CHI '18*. ACM, 185:1–185:15. https://doi.org/10.1145/3173574.3173759

[2] Rahul Arora, Rubaiat Habib Kazi, Fraser Anderson, Tovi Grossman, Karan Singh, and George Fitzmaurice. 2017. Experimental Evaluation of Sketching on Surfaces in VR. In *Proc CHI '17*. ACM, 5643–5654.

[3] 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 - UIST '08*. ACM Press. https://doi.org/10.1145/1449715.1449740

[4] Ravin Balakrishnan, George Fitzmaurice, Gordon Kurtenbach, and Karan Singh. 1999. Exploring interactive curve and surface manipulation using a bend and twist sensitive input strip. In *Proceedings of the 1999 symposium on Interactive 3D graphics - SI3D '99*. ACM Press. https://doi.org/10.1145/300523.300536

[5] Frederick P. Brooks. 1999. What's real about virtual reality? *IEEE Computer Graphics and Applications* 19, 6 (1999), 16–27. https://doi.org/10.1109/38.799723

[6] Xiang Cao and Shumin Zhai. 2007. Modeling human performance of pen stroke gestures. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07*. ACM Press. https://doi.org/10.1145/1240624.1240850

[7] Bruno R. De Araùjo, Géry Casiez, and Joaquim A. Jorge. 2012. Mockup Builder: Direct 3D Modeling on and Above the Surface in a Continuous Interaction Space. In *Proceedings of Graphics Interface 2012 (GI '12)*. Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 173–180. http://dl.acm.org/citation.cfm?id=2305276.2305305

[8] Jennifer Fernquist, Tovi Grossman, and George Fitzmaurice. 2011. Sketch-sketch revolution: an engaging tutorial system for guided sketching and application learning. In *Proceedings of the 24th annual ACM symposium on User interface software and technology - UIST '11*. ACM Press. https://doi.org/10.1145/2047196.2047245

[9] Matthew Flagg and James M. Rehg. 2006. Projector-guided painting. In *Proceedings of the 19th annual ACM symposium on User interface software and technology - UIST '06*. ACM Press. https://doi.org/10.1145/1166253.1166290

[10] Timo Fleisch, Gino Brunetti, Pedro Santos, and André Stork. [n. d.]. Stroke-input methods for immersive styling environments. In *Proceedings Shape Modeling Applications, 2004*. IEEE. https://doi.org/10.1109/smi.2004.1314514

[11] Wei Gai, Cheng Lin, Chenglei Yang, Yulong Bian, Chia Shen, Xiangxu Meng, Lu Wang, Juan Liu, Mingda Dong, and Chengjie Niu. 2017. Supporting Easy Physical-to-Virtual Creation of Mobile VR Maze Games. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*. ACM Press. https://doi.org/10.1145/3025453.3025494

[12] Cindy Grimm and Pushkar Joshi. 2012. Just DrawIt: A 3D Sketching System. In *Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling (SBIM '12)*. Eurographics Association, Goslar Germany, Germany, 121–130. http://dl.acm.org/citation.cfm?id=2331067.2331084

[13] 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 Changing our world, changing ourselves - CHI '02*. ACM Press. https://doi.org/10.1145/503376.503398

[14] Sean Gustafson, Daniel Bierwirth, and Patrick Baudisch. 2010. Imaginary Interfaces: Spatial Interaction with Empty Hands and Without Visual Feedback. In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology (UIST '10)*. 3–12. https://doi.org/10.1145/1866029.1866033

[15] Emmanuel Iarussi, Adrien Bousseau, and Theophanis Tsandilas. 2013. The drawing assistant: automated drawing guidance and feedback from photographs. In *Proceedings of the 26th annual ACM symposium on User interface software and technology - UIST '13*. ACM Press. https://doi.org/10.1145/2501988.2501997

[16] Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka. 1999. Teddy: a sketching interface for 3D freeform design. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques - SIGGRAPH '99*. ACM Press. https://doi.org/10.1145/311535.311602

[17] IKEA. 2016. IKEA Launches Pilot Virtual Reality (VR) Kitchen Experience for HTC Vive on Steam. Blog. Retrieved August 24, 2017 from http://www.ikea.com/us/en/about_ikea/newsitem/040516_Virtual-Reality.

[18] Brent Edward Insko. 2001. *Passive haptics significantly enhances virtual environments*. Ph.D. Dissertation. University of North Carolina at Chapel Hill.

[19] Bred Jackson and Daniel F. Keefe. 2011. Sketching Over Props: Understanding and Interpreting 3D Sketch Input Relative to Rapid Prototype Props. In *IUI '11 Sketch Recognition Workshop*. https://doi.org/10.1145/2047196.2047245

[20] Bret Jackson and Daniel F. Keefe. 2016. Lift-Off: Using Reference Imagery and Freehand Sketching to Create 3D Models in VR. *IEEE Transactions on Visualization and Computer Graphics* 22, 4 (April 2016), 1442–1451. https://doi.org/10.1109/tvcg.2016.2518099

[21] J. Adam Jones, J. Edward Swan, II, Gurjot Singh, Eric Kolstad, and Stephen R. Ellis. 2008. The Effects of Virtual Reality, Augmented Reality, and Motion Parallax on Egocentric Depth Perception. In *Proceedings of the 5th Symposium on Applied Perception in Graphics and Visualization (APGV '08)*. ACM, New York, NY, USA, 9–14. https://doi.org/10.1145/1394281.1394283

[22] Daniel F. Keefe, Robert C. Zeleznik, and David H. Laidlaw. 2007. Drawing on Air: Input Techniques for Controlled 3D Line Illustration. *IEEE Transactions on Visualization and Computer Graphics* 13, 5 (Sept. 2007), 1067–1081. https://doi.org/10.1109/tvcg.2007.1060

[23] Krzysztof Jakub Kruszyński and Robert van Liere. 2009. Tangible props for scientific visualization: concept, requirements, application. *Virtual Reality* 13, 4 (July 2009), 235–244. https://doi.org/10.1007/s10055-009-0126-1

[24] Manfred Lau, Greg Saul, Jun Mitani, and Takeo Igarashi. 2010. Modeling-in-context: user design of complementary objects with a single photo. In *Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium*. Eurographics Association, 17–24. https://doi.org/10.2312/sbm/sbm10/017-024

[25] Jérémy Laviole and Martin Hachet. 2012. PapARt: Interactive 3D graphics and multi-touch augmented paper for artistic creation. In *2012 IEEE Symposium on 3D User Interfaces (3DUI)*. IEEE, 3–6. https://doi.org/10.1109/3dui.2012.6184167

[26] Pedro Lopes, Sijing You, Lung-Pan Cheng, Sebastian Marwecki, and Patrick Baudisch. 2017. Providing Haptics to Walls & Heavy Objects in Virtual Reality by Means of Electrical Muscle Stimulation. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*. ACM Press. https://doi.org/10.1145/3025453.3025600

[27] William A. McNeely. [n. d.]. Robotic graphics: a new approach to force feedback for virtual reality. In *Proceedings of IEEE Virtual Reality Annual International Symposium*. IEEE. https://doi.org/10.1109/vrais.1993.380761

[28] Paul Milgram and Fumio Kishino. 1994. A Taxonomy of Mixed Reality Visual Displays. *IEICE Transactions on Information Systems* E77-D (1994). http://vered.rose.utoronto.ca/people/paul_dir/IEICE94/ieice.html

[29] Donald A. Norman. 2013. *The Design of Everyday Things*. Basic Books, Inc., New York, NY, USA.

[30] Luke Olsen, Faramarz F. Samavati, Mario Costa Sousa, and Joaquim A. Jorge. 2009. Sketch-based modeling: A survey. *Computers & Graphics* 33, 1 (Feb. 2009), 85–103. https://doi.org/10.1016/j.cag.2008.09.013

[31] Chris De Paoli and Karan Singh. 2015. SecondSkin: sketch-based construction of layered 3D models. *ACM Transactions on Graphics* 34, 4 (July 2015), 126:1–126:10. https://doi.org/10.1145/2766948

[32] Robert Pastel. 2006. Measuring the Difficulty of Steering Through Corners. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, New York, NY, USA, 1087–1096. https://doi.org/10.1145/1124772.1124934

[33] Huaishu Peng, Jimmy Briggs, Cheng-Yao Wang, Kevin Guo, Joseph Kider, Stefanie Mueller, Patrick Baudisch, and François Guimbretière. 2018. RoMA: Interactive Fabrication with Augmented Reality and a Robotic 3D Printer. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 579, 12 pages. https://doi.org/10.1145/3173574.3174153

[34] Alec Rivers, Andrew Adams, and Frédo Durand. 2012. Sculpting by numbers. *ACM Transactions on Graphics* 31, 6 (Nov. 2012), 157:1–157:7. https://doi.org/10.1145/2366145.2366176

[35] Johannes Schmid, Martin Sebastian Senn, Markus Gross, and Robert W. Sumner. 2011. OverCoat: an implicit canvas for 3D painting. In *ACM SIGGRAPH 2011 papers on - SIGGRAPH '11*. ACM Press. https://doi.org/10.1145/1964921.1964923

[36] Ryan Schmidt, Azam Khan, Karan Singh, and Gord Kurtenbach. 2009. Analytic drawing of 3D scaffolds. *ACM Transactions on Graphics* 28, 5 (Dec. 2009), 149:1–149:10. https://doi.org/10.1145/1618452.1618495

[37] Hyunyoung Song, François Guimbretière, Chang Hu, and Hod Lipson. 2006. ModelCraft: capturing freehand annotations and edits on physical 3D models. In *Proceedings of the 19th annual ACM symposium on User interface software and technology - UIST '06*. ACM Press. https://doi.org/10.1145/1166253.1166258

[38] Christian Weichel, Jason Alexander, Abhijit Karnik, and Hans Gellersen. 2015. SPATA: Spatio-Tangible Tools for Fabrication-Aware Design. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction - TEI '14*. ACM Press. https://doi.org/10.1145/2677199.2680576

[39] Christian Weichel, Manfred Lau, David Kim, Nicolas Villar, and Hans W. Gellersen. 2014. MixFab: a mixed-reality environment for personal fabrication. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*. ACM Press. https://doi.org/10.1145/2556288.2557090

[40] Gerold Wesche and Hans-Peter Seidel. 2001. FreeDrawer: a free-form sketching system on the responsive workbench. In *Proceedings of the ACM symposium on Virtual reality software and technology - VRST '01*. ACM Press. https://doi.org/10.1145/505008.505041

[41] 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 (SBIM '10)*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 135–142. http://dl.acm.org/citation.cfm?id=1923363.1923387

[42] Min Xin, Ehud Sharlin, and Mario Costa Sousa. 2008. Napkin sketch: handheld mixed reality 3D sketching. In *Proceedings of the 2008 ACM symposium on Virtual reality software and technology - VRST '08*. ACM Press. https://doi.org/10.1145/1450579.1450627

[43] 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 Transactions on Graphics* 33, 4 (July 2014), 1–13. https://doi.org/10.1145/2601097.2601128

[44] Ya-Ting Yue, Xiaolong Zhang, Yongliang Yang, Gang Ren, Yi-King Choi, and Wenping Wang. 2017. WireDraw: 3D Wire Sculpturing Guided with Mixed Reality. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*. ACM Press. https://doi.org/10.1145/3025453.3025792

[45] Kening Zhu, Alexandru Dancu, and Shengdong (Shen) Zhao. 2016. FusePrint: A DIY 2.5D Printing Technique Embracing Everyday Artifacts. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems - DIS '16*. ACM Press. https://doi.org/10.1145/2901790.2901792