

DesignAR: Immersive 3D-Modeling Combining Augmented Reality with Interactive Displays

Patrick Reipschläger

Interactive Media Lab Dresden
Technische Universität Dresden, Germany
patrick.reipschlaeger@tu-dresden.de

Raimund Dachsel

Interactive Media Lab Dresden
Technische Universität Dresden, Germany
dachsel@acm.org

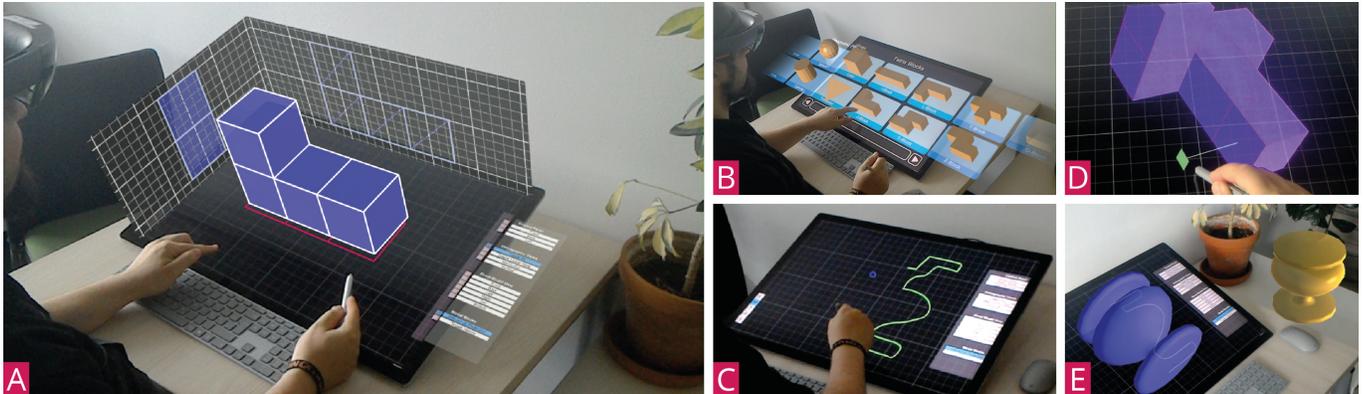


Figure 1: A general overview of *DesignAR*, highlighting (a) the augmented workstation as a whole, (b) The AR object browser, (c) sketching contours, (d) modeling by extruding faces, and (e) rotational solids and AR instances.

ABSTRACT

We present *DesignAR*, an augmented design workstation for creating 3D models. Our approach seamlessly integrates an interactive surface displaying 2D views with head-mounted, stereoscopic Augmented Reality (AR). This creates a combined output space that expands the screen estate and enables placing 3D objects beyond display borders. For the effective combination of 2D and 3D views, we define different levels of proximity and alignment. Regarding input, multi-touch and pen mitigate issues of precision and ergonomics commonly found in mid-air VR/AR interaction. For creating and refining 3D models, we propose a set of pen and touch techniques with immediate AR feedback, including sketching of rotational solids or tracing physical objects on the surface. To further support a designer's modeling process, we additionally propose orthographic model views and UI offloading in AR as well as freely placeable model instances with real-world reference. Based on our *DesignAR* prototype, we report on challenges and insights regarding this novel type of display augmentation. The combination of high-resolution, high-precision interactive

surfaces with carefully aligned AR views opens up exciting possibilities for future work and design environments, a vision we call *Augmented Displays*.

Author Keywords

Augmented Reality, Interactive Surfaces, Augmented Displays, Touch+Pen Interaction, Design, 3D Modeling.

CCS Concepts

•Human-centered computing → Mixed / augmented reality; Interactive systems and tools; Interaction techniques; Touch screens; Gestural input;

INTRODUCTION

Computer-generated images and consequently 3D models have pervaded nearly all areas of our modern life. Besides typical fields like computer games, advertisement, and film, the emerging fabrication and 3D printing communities have added another demand for 3D models. However, most conventional modeling applications remain firmly rooted in desktop environments. These existing commercial tools provide only limited support for natural interaction, although the literature suggests the benefits of multi-touch and pen input for the design process [6, 27, 21]. Similarly, support for stereoscopic or spatial representation of modeled 3D objects is lacking. As a result, the potential of users interacting with virtual 3D models with the same naturalness as with real objects remains unrealized. Recently, Mixed Reality techniques and devices have been on a constant rise. Researchers have become increasingly

Permission to make digital or hard copies of part or all 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 third-party components of this work must be honored. For all other uses, contact the owner/author(s).

ISS'19, Proceedings of the 2019 ACM International Conference on Interactive Surfaces and Spaces, November 10–13, 2019, Daejeon, Republic of Korea

© 2019 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-6891-9/19/11.

DOI: <https://doi.org/10.1145/3343055.3359718>

interested in Virtual or Augmented Reality environments for creating 3D content, and there even exist commercial products like Google TiltBrush [2]. While these tools provide a high grade of immersion in the perception of 3D objects, they particularly lack means of precise interaction. Most allow the creation of 3D curves or objects directly in 3D space, for instance by using mid-air gestures. This results in a general absence of precision and physical constraints that are required for precise work with a high degree of detail [5].

In contrast to a fully virtual environment, we propose a traditional workstation typically used for design tasks, such as a high-resolution interactive surface, and extend it with head-coupled Augmented Reality (AR). This mitigates the lack of precision of, e.g., mid-air gestures, by utilizing the precise input provided by the interactive surface while at the same time enabling the advantages of displaying 3D models in stereoscopic AR. We present *DesignAR*, an AR environment for modeling simple 3D objects, which combines a tilted interactive high-resolution display with multi-touch and pen input with an AR head-mounted display (HMD) (see Figure 1). The 3D model is viewed through the HMD and situated in AR directly on the display while the display is mainly used for interaction and for showing additional information. For now, we focus on the modeling of simple, single objects instead of complex 3D scenes to illustrate the benefits of our concepts and advantages of our setup over traditional workstations. Consequently, our contributions are as follows:

- Our ***DesignAR* augmented workstation**, combining a head-coupled, stereoscopic AR view of a 3D model with precise pen and touch input of an interactive surface, and an additional exploration of different levels of proximity of AR objects in relation to the display.
- A set of pen and touch **modeling interaction techniques** for creating and refining simple objects in AR.
- **Orthographic AR projections** on the display borders, which provide users with dedicated spatially aligned 2D views while working directly with the 3D model.
- AR instances of the modeled objects with **real world reference** directly in the user's environment to strengthen the connection between the virtual and real world.
- A working **prototype** and a discussion of the challenges we faced and the insights gained.

In the following, after an overview of related work, we subsequently elaborate on these contributions in a similar order, before discussing our prototype and the technical and usability challenges we encountered. Furthermore, we contribute our vision of *Augmented Displays*, a new class of display systems which seamlessly combines interactive surfaces with head-coupled Augmented Reality.

RELATED WORK

DesignAR and its presentation and interaction techniques touch many areas of HCI. Thus, there is a lot of prior work that is important for particular aspects of this paper. In the following, we first give a structured overview of work on creating 3D objects in general before we look more deeply at papers that have a stronger relation to immersive modeling and *DesignAR*.

Afterwards, we describe work concerning the distribution of user interfaces as a more loosely related field.

Enhancing the creation of 3D objects by using immersive technologies has a long history in HCI research, from very early works by Butterworth et al. [13], Deering [17], and Wesche and Seidel [54] to very recent ones for free form sketching of curvatures [29, 5] and surface modeling [37, 30]. Even commercially developed tools like Google TiltBrush [2] are available. For the scope of this work, we chose to exclude outdoor systems like work by Piekarski [44] or Zlatanova [58] and focus on indoor design environments. In almost all related work, a rough distinction can be made between the two main categories of 3D sketching [5, 18, 29, 32, 33, 34, 55], i.e., drawing lines or contours, and 3D modeling [9, 15, 30, 38, 42, 43, 47, 49, 57], i.e., creating polygonal or parametric surface models. Of those research projects that employ some form of Mixed Reality, most use Virtual Reality [32, 29, 37, 38, 42, 47, 57], some use Augmented Reality [5, 15, 18, 49], and even unconventional technologies like Light Field Displays [50]. The used display technologies are very diverse. Many tasks, especially in the VR area, rely on Head-Mounted-Displays [7, 18, 37, 38]. In addition to general stereoscopic displays [9, 32], specialized display solutions [24, 35, 53] are also represented. Mobile devices [4, 49, 55] or interactive surfaces [5, 34, 43] are used as well, sometimes even in combination with HMDs [11]. The input modalities are similarly diverse and range from freehand interaction [9, 15, 18, 35, 38] over tracked 6 DoF devices [37, 42, 47, 57] to pen [5, 32, 55] and touch input [5, 34, 49]. Even special haptic input devices [7] are used.

This illustrates the richness and breadth of related work within the field of immersive 3D modeling. Talking about specific works in detail, *Toucheo* by Hachet et al. [24] use a specialized, stereoscopic display solution in combination with a touch screen to enable the manipulation of 3D objects. While providing easy 9 DoF interaction for novice users, the application offers no modeling or sketching and no interaction beyond display borders. *Mockup Builder* by De Araújo et al. [15] features free-hand tracking above a touch tabletop and a stereoscopic view using a projector and shutter glasses for building 3D models. *SpaceTop* by Lee et al. [35] lets users interact through a video-see-through display and supports interacting with 2D and 3D content but is neither stereoscopic nor does it allow 3D content to be placed above or around the display. Mine et al. [42] combines modeling in VR with a tracked smartphone. The mobile device is used for pointing and widget interaction, but there is no strong spatial relation between the screen and the manipulated objects. *MixFab* by Weichel et al. [53] combines projected Augmented Reality with free hand interaction for creating 3D objects out of geometric primitives for fabrication. Focusing on 3D printing, the input and output volume is confined to the dimensions of a typical printer bed. Tang and Duong [49] propose creating 3D models by combining geometric primitives using AR and touch interaction on a mobile phone. *DualCAD* by Millette and McGuffin [41] provides both a traditional desktop CAD environment with mouse and keyboard input as well as head-mounted AR using a mobile device for pointing and input. However, while users may freely switch between both modes, the HMD and the

desktop are not fully integrated into a seamless system and do not use natural touch input. *SymbiosisSketch* by Arora et al. [5] combines free hand drawing and drawing on surfaces to create objects in Augmented Reality. Users can define planes in AR, on which they then draw using a tablet with pen interaction. However, the interaction on the tablet is spatially decoupled from the AR surface that is sketched onto. Kim et al. [33] propose using hand motions to create scaffolds, that can be used for sketching non-polygonal 3D objects via pen input on a non-stereoscopic interactive surface. Lastly, there are conventional tools like *Shapr3d* [4], which enable designing CAD models on a mobile device but without using stereoscopic or MR components. Additionally, *grib* [3] enables creating 3D models using AR on a mobile phone in a similar way to Tang and Duong [49]. Furthermore, Autodesk and Microsoft presented a demo for using the HoloLens in combination with Maya [1] but no subsequent release followed.

Besides 3D modeling and sketching, distributed user interfaces are another established research area relevant to *DesignAR*. Distributed user interfaces can be defined as user interfaces whose components are distributed across one or more of the dimensions input, output, platform, space, and time [19]. Early examples are *Hybrid User Interfaces* by Finer and Shamash [20], *Augmented Surfaces* [45], and the *Emmie* system [14]. More recent work by Benko et al. [8] investigates interaction techniques for a collaborative, heterogeneous workspace consisting of a 2D multi-touch projection surface, 3D content provided by HMDs, and spatially tracked gloves. *IllumiRoom* [31] augments the surroundings of a TV with projected visualizations, thereby extending the display space. *MultiFi* by Grubert et al. [23] proposes distributed user interfaces among multiple devices like HMDs, smartwatches, and smartphones. Butscher et al. [12] combine an interactive tabletop with a video-see-through AR HMD for analyzing 2D visualizations. These visualizations are situated in AR above the tabletop and users can interact with them using a proxy visualization on the tabletop. *PerfVis* by Merino et al. [39] provides user awareness for software performance by placing AR 3D visualizations next to a conventional display. Although there is a semantic connection between content on the display and the AR visualization, there is no spatial connection. This is similar to the work by Sereno et al. [48], where an AR 3D visualization can be manipulated through touch on a tablet that shows an orthographic view of a dataset. *HoloDoc* by Li et al. [36] proposes to use AR documents in conjunction with paper documents in an augmented desk setup. The work uses a combination of mid-air gestures and pen input for interaction.

In summary, the connection, especially the spatial one, between display and Mixed or Augmented Reality content is often only loosely described in the literature. The augmentation of objects directly on the screen is uncommon. Work in this regard almost always concentrates strictly on the display or surface screen itself without considering the space around the display at all. Touch or pen input is rarely used to interact with objects in stereoscopic Augmented Reality in a direct manner. Instead, object manipulation is either achieved with widgets or proxies or through a non-stereoscopic mobile device.

GENERAL DESIGNAR CONCEPT

The central idea of *DesignAR* is to create an augmented workstation by extending a traditional modeling and design environment with AR capabilities. We achieve this by seamlessly integrating a tilted, multi-touch and pen enabled interactive surface with a head-mounted, stereoscopic Augmented Reality display, creating a combined output and interaction space. Thus, we address the lack of immersion of traditional displays and gain the ability to place 3D objects above and beyond the screen, vastly expanding the available display space. At the same time, we mitigate the disadvantages of free-form 3D sketching and modeling applications using the constraints and natural pen and multi-touch input of the interactive surface.

Our *DesignAR* setup has some similarities to the well known fish-tank metaphor [52]. However, by using an AR HMD, there are no cut-off issues at the edges of the display, and the model can extend even beyond the screen borders. In contrast to other work that combines AR with conventional displays, like Butcher et al. [12] or Marino et al. [39], we place special emphasis on the alignment of the AR content in relation to the display. Our goal is to create a seamless transition between content in AR and on the display to invoke the impression of a single, coherent system. Therefore, the AR space around the display acts as a natural extension of the display itself. Furthermore, we use an off-the-shelf design oriented interactive surface with a high-resolution display and pen and multi-touch input. This results in a setup that closely resembles typical workstations useful for design work instead of a specialized research setup.

Proximity of AR Content to the Display

The placement of AR content in relation to the display strongly influences the impression of a seamlessly integrated system. Instead of placing AR objects somewhere in 3D space, we differentiate between three levels of spatial proximity of AR content in relation to the display. These levels represent the degree of coupling between the AR objects and content shown on the display, which we devised in the context of *DesignAR*.

- L1** The strongest of these connections is between the display and the modeled 3D object, which occupies the space directly in front of or behind the display and is perfectly aligned to the orthogonal projection on the display itself (see Figure 1a center).
- L2** AR content which is arranged at the edge or close to the edge of the display still bears a clear connection to the display itself. The AR objects can still be spatially aligned as extensions of display content, like the offloaded menus (see Figure 6), or along the main axes of the display like the orthographic views (see Figures 1a and 5).
- L3** The weakest connection is between the display and AR content that no longer shows a clear spatial relation or alignment to the display. This applies to AR instances of the modeled 3D object which have been freely placed in 3D space (see Figure 8). Since these representations change together with the modeled object, there is still a clear semantic connection. However, this connection is purely virtual and no longer determined by spatial proximity or orientation.

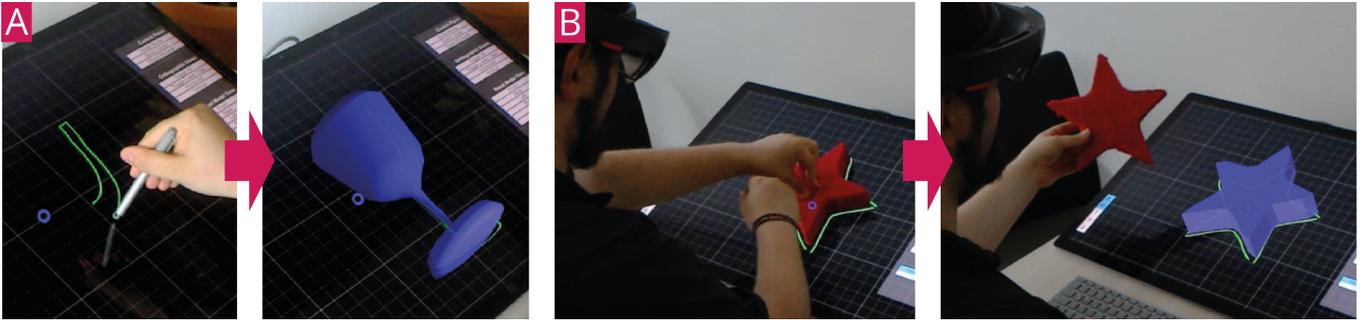


Figure 2: Examples of *DesignAR*'s different options for creating new models: (a) sketching a contour to create a rotational solid and (b) tracing the contour of a real world object, which is then extruded.

Interaction Principles

DesignAR utilizes bimanual touch and pen input for interaction within **L1**. A frame around the display's border is reserved for touch interaction with AR objects in **L2**. For all content in **L3**, we choose to use mid-air interaction. In **L1**, every action that introduces new geometry to the model is done by pen. Other actions, like selection or view manipulation, are achieved by multi-touch input (comparable to Hinckley et al. [27]). The decision for using a pen for modeling has two reasons: First, this avoids the fat finger problem as the much finer tip of the pen allows for more precise interaction, like sketching or view manipulation. Second, it builds on the mental model of the pen as a dedicated tool for creating new content. Similar to creating new strokes by drawing on paper, the pen creates new geometry within the model. Touch input uses either a one finger or two finger drag for many operations: Whenever the user drags an object with one finger, the interaction affects the x and y axis of the display. Using two fingers affects the perpendicular z axis. We want to emphasize that the axes of the display are used for every interaction instead of the local axes of the model itself. Thus, the gestures a user performs on the display conform with the resulting transformation of the model.

MODELING FUNCTIONALITY

In order to meet the demands of an immersive modeling environment, creating and manipulating models is essential. Instead of complex scenes, we decided to focus on the modeling of simple, single objects to illustrate the core principles of our concepts. However, our concepts are also applicable to more complex use cases or even commercial tools. Similarly, our interaction techniques aim for a set of elementary functions and can also be extended by more sophisticated techniques from the literature, e.g., like in *Teddy* [28] or *ILoveSketch* [6]. We divided the following section in two parts: First, we describe how to create new models and second, we explain how to refine them afterwards.

Creating New Models

To present users with flexibility suitable to pursuing their current design goals, *DesignAR* provides several ways to create new models.

We first propose a 3D object browser which presents a catalog of available 3D models to users. This includes geometric primitives, such as cubes, spheres, and cylinders, as well as stored objects from previous sessions (see Figure 1b). 3D objects above the display are viewed through stereoscopic AR to allow a better spatial impression for users when selecting objects. A touch scrollbar enables browsing all objects and a preview of previous and upcoming entries is displayed adjacent to the display in AR. Users select an object with a simple tap on the corresponding entry and *DesignAR* switches to the default modeling view, where the model can be refined.

The second method we propose is sketching a contour with the pen (see Figure 2a). The contour is converted into a Bezier spline, whose support points can be adjusted using touch. Based on the spline, a polygonal rotational solid is instantly created in AR space, thereby allowing users to directly observe the result of their sketching. Changes to the contour immediately affect the rotational solid to enable quick adjustments of the object's shape. After sketching a satisfactory object, users can change to the default modeling view to refine it further.

As a third variant, we propose to create the model based on a real object, which is placed on the surface of the interactive display to trace its contours with the pen. The resulting outline is then extruded in the direction of the z axis. Users have a short period of time (e.g., 2 seconds) to start a two finger drag gesture to manipulate the height of the extrusion. After the time elapsed or the gesture is finished, a polygonal model is created in AR space, which can then be further refined.

Refining Models

Besides creating new models from scratch, functions to refine existing models are equally important. *DesignAR* utilizes a Box Modeling approach, where a generic primitive, like a box, is successively divided into smaller faces until the desired shape is achieved (i.e., subdivision modeling). *DesignAR* supports a set of pen and multi-touch techniques that are designed to derive their meaning from the context they are performed in to avoid mode switches:

1. Users can *move vertices* on all three axes by a small handle, which can be dragged on the x- and y-axis with one finger and on the z-axis with two fingers. To mitigate the fat finger problem, the pen can also be used.

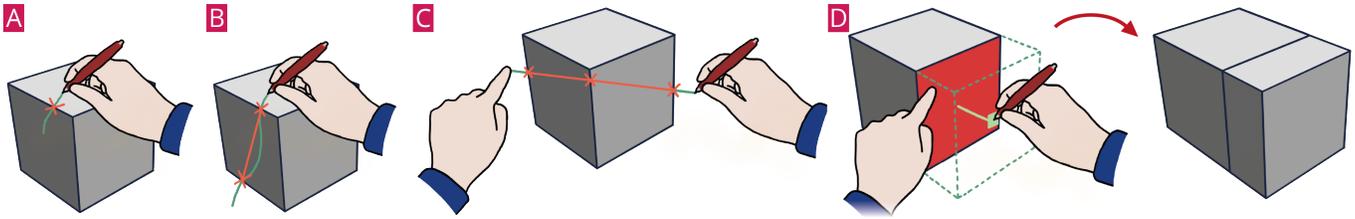


Figure 3: Sketches illustrating our interaction techniques for refining 3D models: (a) creating a new vertex by crossing a single edge with a pen stroke, (b) creating a new edge by crossing two or more edges with a single pen stroke, (c) slicing the whole model by using pen and touch to define a slicing plane, (d) extruding a face by first selecting it with the finger and then dragging it out with the pen.

2. To **create new vertices**, users draw a line with the pen across an existing edge. This splits the edge into two at the intersection (see Figure 3a).
3. To **create a new edge**, users draw a line that crosses two edges of a visible polygon. Vertices are created at the intersections and a new edge is created between those vertices. The polygon is consequently split into two (see Figure 3b).
4. A **slice** through the whole geometry can be created by using one finger and the pen to define a line (see Figure 3c). Every visible and hidden polygon the line crosses is split into two and new vertices and edges are created at the intersection.
5. Users can **extrude** a polygon along its normal vector. The polygon is touched with one finger and a line is drawn with the pen to determine the amount of extrusion (see Figure 3d). Moving the pen adjusts the extrusion and a wireframe provides feedback. Lifting the finger completes the interaction, pushes the polygon outwards, and creates a new quad for each edge of the polygon.

Previous research [10, 51, 22] found that precise touch interaction with stereoscopic 3D objects can be difficult because of a perceptual mismatch. Furthermore, since the 3D models in DesignAR are situated above the tabletop, users have to sometimes reach through the model itself to touch the display, which makes the interaction hard to see and can feel unnatural. To address these issues, a 2D projection of the model on the display can be used when interacting. Whenever users start a modelling operation, we propose to switch to this 2D projection and as soon as the operation is finished, switch back to the normal, stereoscopic AR view of the model (see Figure 4). Using the spatial position of the HMD in relation to the dis-

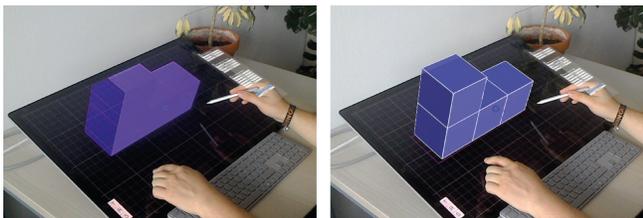


Figure 4: When a user starts a modeling operation by touching the screen, DesignAR switches from the AR representation to a 2D projection of the model from the users perspective displayed on the screen (left). When the interaction is finished, the model switches back to the stereoscopic AR version (right).

play, a very accurate head-coupled, off-center projection can be calculated, which closely resembles the current perspective of the 3D model as seen through the AR headset. This reduces the mental load of transiting between both views. The 2D projection on the display temporarily sacrifices the stereoscopic rendering of the AR HMD. The 3D object is also clipped at the display’s border. However, users can now interact directly with the 2D projection, and the combination of pen and touch with the display’s high resolution allows for very precise interaction, which is important for many operations like slicing and creating edges.

NAVIGATION AND TRANSFORMATION

Facilitating fast and efficient design capabilities requires, besides good modeling functionality, easy control of the current viewport. DesignAR uses dedicated toggle buttons on the side of the display to switch between transformation modes. Using bimanual interaction, users switch between translation, rotation, and scaling modes with the non-dominant hand and perform the corresponding transformation with the dominant hand. Using separate modes for each transformation enables us to employ simple one and two finger drag gestures users already know from other parts of the prototype. We explored previous work for transforming 3D objects using touch gestures (e.g., [16, 25]), but we deliberately concentrated on simple interaction and separated degrees of freedom for better precision and control. This also reduces the mental demand as it is difficult to find expressive gestures for all interaction modes that are easy to remember and to execute, and at the same time do not overlap with other interaction techniques. Work by Yu et al. [56] proposes to use the borders of the screen for the different DoFs, but DesignAR already uses them to interact with AR content.

For the sake of consistency, manipulating the viewport employs our aforementioned general interaction concept and uses multi-touch with one or two finger gestures. The model is moved on the x and y axis, i.e., sideways and upwards on the display, by dragging with one finger in the corresponding direction. It is moved on the z axis, i.e., above and below the display, by dragging with two fingers. The same principle applies to rotation and scaling. Additionally, we use a pinch gesture to scale the model uniformly as it is a well established gesture for zooming and therefore suitable for the comparable operation of scaling an object. All transformation operations are performed on the axes of the display, not the local axes of

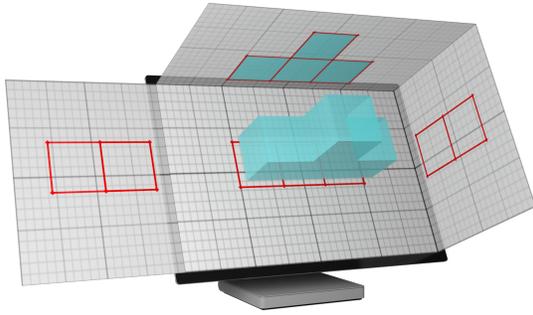


Figure 5: 3D rendering illustrating *DesignAR's* orthographic AR views on the borders of the display, showing individual 2D projections of the modeled object. The left view is in a folded position to increase viewability and the render mode of the top view was changed from wireframe to solid.

the model, to ensure a strong connection between the gestures and the resulting actions. For instance, a drag to the left will always rotate the model leftwards in the same direction even if that means simultaneously rotating around multiple local axes of the model. To quickly rotate around the main axes of the display in steps of 90 degrees, we propose using a five finger swipe gesture from the display border to the center of the screen. For instance, when a user performs a five finger downward swipe, the model rotates 90 degrees downward (i.e., around the x axis) as well. All orthographic projections are updated accordingly.

ADDITIONAL MODELING SUPPORT

To further support the user in designing models, we propose a number of additional features that take advantage of *DesignAR's* augmented workstation. Besides the modeled object, we suggest to place additional AR content at the edges of the display. The AR content maintains a strong connection to the display while at the same time gaining additional visualization space.

Orthographic Projections

Two-dimensional orthographic wireframe views belong to the standard repertoire of most 3D modeling applications. They provide a dedicated view from the main axes of a coordinate system and reduce the dimensionality of a 3D model. Therefore, orthographic views are a useful tool for understanding and editing complex geometry. They are often used in a 2x2 tiling of the screen, providing additional views of the 3D model, but severely reducing the screen space of a single view. The views can commonly be zoomed and panned independently and bear no immediate spatial relation to other views. Although changes to the model immediately update all other views, it can still be challenging for a user to understand which section of a model the orthographic view displays.

To address these issues, we propose to place the orthogonal views directly on the edges of the display in AR space (see Figures 1a and 5). In contrast to traditional solutions, this does not sacrifice any screen estate at all. Each view is placed

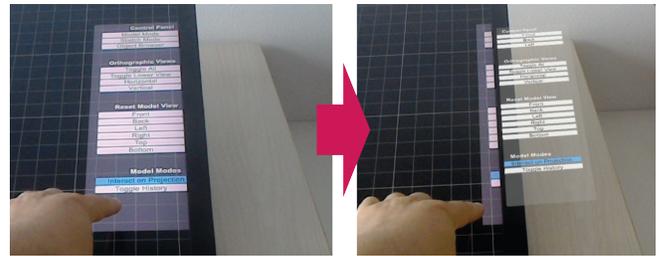


Figure 6: The menu situated on the display is offloaded into AR space with a swipe. The representative handles on the display's border can be used to interact with the offloaded menu.

on one main axis of the display and shows a corresponding cross-section of the model. This makes it immediately obvious which part of the model is shown in each orthographic view, which is not the case in a traditional 2x2 grid. All views are always synchronized with the 3D model view, so that moving, rotating, and scaling the model affects all views. However, this synchronization prevents the independent zooming or panning of individual views. In addition to the AR orthographic views on the display's border, a top view is visualized on the display's screen itself, thereby making use of the display's high resolution.

The orthographic views may exhibit potential perception problems when viewed from uncomfortable angles. Therefore, we propose that the views can be seamlessly folded at the edges of the display (see Figure 5 left side). This results in a more planar arrangement, which weakens the mental model of the view cube but eliminates any perception problems. To fold a view, users perform a pinch-gesture on the edge of the display of the orthographic view they want to flip. An opening pinch seamlessly rotates the view outwards to a more planar orientation, while a closing pinch rotates it inward until it is perpendicular to the display. The visibility of individual views can be toggled by performing a double tap at the corresponding display edge, enabling users to select only the views they are interested in. Additionally, users can switch between wireframe and solid rendering modes of individual views by performing a swipe with one finger on the corresponding display border. Additional render modes can be accessed through a conventional menu (see Figure 6).

UI Offloading and Interaction with AR Menus

Complex tools like modeling applications often contain convoluted menus and toolbars that use up valuable screen space, which could instead be used for the modeling workspace itself. Although the need for large menus can be reduced by using appropriate gestures and interaction concepts, menus cannot be eliminated completely for sufficiently complex tools.

In order to maximize the available area for modeling, we propose foldable menus. In contrast to conventional collapsible menus, we suggest moving the menus to AR space instead of hiding them. With a one finger swipe gesture performed on the menu panel towards the display's edge, the entire menu moves into the adjacent AR space (see Figure 6). Vice versa,

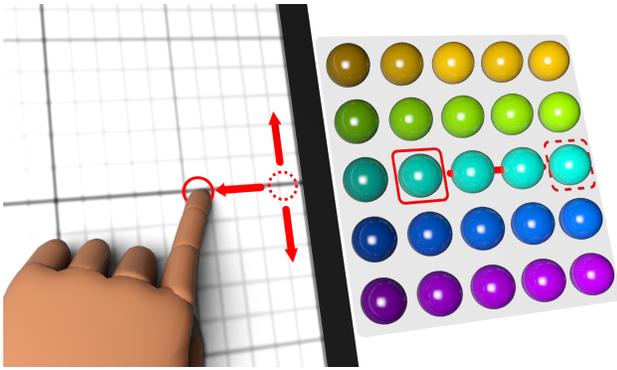


Figure 7: 3D rendering illustrating the two-dimensional interaction with offloaded AR menu items in a grid layout using the example of a material catalogue with 3D spheres.

another swipe from the edge of the display transfers the menu back to the display. While the function and current state of buttons, toggles, and checkboxes remains visible even when collapsed, it is not possible to interact with an offloaded menu using touch input as they are no longer situated on the display. To address this drawback, we propose to use little handles, just big enough to be comfortably touchable, at the border of the display (L2), where the former menu-items were located. Thus, only very little actual screen space is sacrificed and the menu remains interactive.

While the aforementioned technique only provides one dimensional interaction, we propose a two dimensional method as well, suitable for browsing large collections of objects or options. The objects are situated next to the display in a similar fashion as above, but in a grid layout (see Figure 7). Thus, the available AR space besides the display allows the preview of 3D objects in head-coupled, stereoscopic perspective. Users can interact with them by touching the border of the screen. This highlights the rightmost object in the corresponding row. Moving the finger leftwards selects the objects to the left in sequential order while moving the finger up or down selects the corresponding objects in the rows above or below. When the finger is lifted, the selected object's action is triggered. While a direct mapping is sensible for vertical selection (rows), we propose a non-uniform mapping for selecting objects within a row to reduce the required finger movement. This technique enables offloading even large quantities of menu objects into AR space, provided they can be represented within a grid.

AR Instances of Modeled Objects

The AR space *DesignAR* provides is not limited to the area in close proximity to the display but extends to the entire environment. Utilizing this, we propose to give users the ability to create AR instances of the currently modeled 3D model that can be freely placed in the environment. These instances are not copies of the 3D object, but references that are instantly updated with any changes made to the 3D model. Besides other use cases, this can be useful for later 3D printing of the model, because it provides a preview how the object looks like within the real environment.

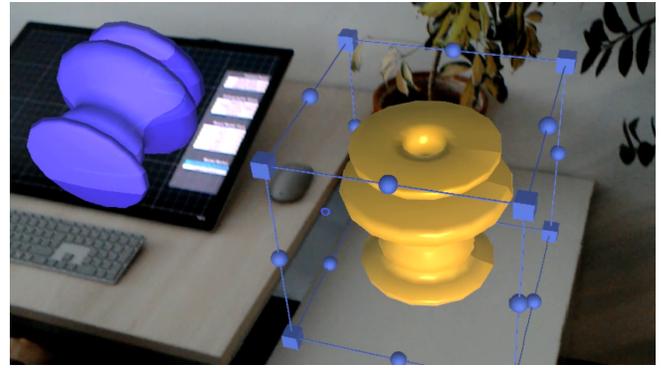


Figure 8: A rotational solid (blue) was created on the display and a 3D instance (yellow) has been placed on the drawer next to it. The transformation widget can be seen around the 3D instance.

Until now, we proposed to only use multi-touch for modeling and interacting with menus or AR objects on the display border, while deliberately avoiding mid-air interaction. However, mid-air interaction is well suited for interacting with AR instances, as they can reside anywhere in 3D space and have no spatial relation to the display (L3). Furthermore, they do not require the same precise input as, e.g., modeling. Therefore, we propose to use an air-tap gesture and a dedicated transformation widget for manipulating the 3D instances (see Figure 8). The widget can be freely dragged on all three axes. Cube-shaped handles on the corners can be used to uniformly scale the instance while circular handles on the edges rotate the model on the corresponding axis. To create a new AR instance, a user simply grabs the object by performing an air tap and drags it outside of the display area. To avoid obstructing the view of the AR instance and its environment, the transformation widget can be hidden (and displayed again) by executing an air tap on the AR instance. To further increase the degree of embeddedness of the AR instance into the real environment, spatial mapping capabilities of the AR headset may be used to hide parts of the 3D instance that are covered by real objects. 3D instances can be removed with a small button at the bottom of the widget or with a dedicated throw-away gesture.

PROTOTYPE

Our previously introduced *DesignAR* concepts were developed in an iterative development process with a tight interplay of conceptual work and prototyping. In addition to our own experience, we also drew on the knowledge of usability experts from our department. We implemented all our presented concepts, except the 2D grid menu interaction. In the following, we describe our setup and technical details.

We used the 28-inch Microsoft Surface Studio as an interactive display, which is targeted at graphic designers. It offers a high resolution of 4500x3000 pixels, supports pen and multi-touch input, and is seamlessly tiltable from an upward position to nearly horizontal, slightly tilted position. For the AR HMD, we decided to use the state-of-the-art Microsoft HoloLens, which possesses good spatial tracking and image quality but suffers from a very small diagonal field-of-view of ca. 30

degrees. The prototype was developed with the Unity 3D engine, running both on the display and the HoloLens. After starting *DesignAR's* HoloLens application, users must first manually adjust the position of the AR content. Because all AR objects with a spatial relation to the display have a common anchor point, this has to be done only once. The anchor is placed on the bottom left corner of the screen using an AR widget that can translate or rotate the object on all three axes. It is similar to 3D gizmos common in most 3D applications and controlled by the HoloLens's air tap. When satisfied with the position, the gizmo can be hidden to not obstruct the rest of the application.

DesignAR consists of two basically independent applications. This poses the challenge of generating a seamless visual impression, which can be fluently interacted with. We developed a couple of techniques to address these issues and strengthen the impression of a seamless systems. Our network infrastructure is mandatory for synchronizing both systems and is based on a client-server architecture using TCP/IP and Open Sound Control (OSC) for exchanging messages. All devices register with a dedicated server where they state if they are either display or HMD and what applications they are running. Each device is assigned a unique session ID, all other devices are notified about the new participant, and may then subscribe to specific events sent by that device. Although *DesignAR* only uses just a single display and HMD, our network supports an arbitrary number of displays and HMDs, which simplifies the future expansion of the system by additional devices.

Most components in *DesignAR* have a representation on the display and the HoloLens side which must be synchronized. A combination of component specific addresses and clearly defined interaction types enables us to effectively dispatch events. Many actions, such as moving the 3D model, are processed independently, after the initial trigger event, by display and HoloLens in order to reduce network traffic. To avoid different states, like the slow drift of the model on one side, both systems are synchronized after the end of the interaction. Thereby, the status of the HoloLens has priority over that of the display. Although the modeled object is only viewed through the HoloLens, an exact copy of it exists on the display. This is used to realize the orthographic projection of the model on the screen. It is also important to calculate the head-coupled projection of the model on the screen whenever a user starts a modeling action. An additional virtual camera is used which is synchronized with the current position and orientation of the HoloLens and uses off-centered projection to correctly render the model from the user's perspective. *DesignAR's* orthographic views are realized through additional cameras looking at the 3D model from the corresponding angles. They render into a texture which is then mapped to a quad positioned on the display's border.

DISCUSSION OF CHALLENGES

Adding to our own experience, we also discussed our prototype with several usability experts from our lab. We will now address the technological and usability challenges we faced and the insights we gained in developing *DesignAR*. One of the central aspects in this respect was how to best evoke the im-

pression of a single, seamless system to achieve a high degree of immersion and provide the best possible user experience. In the following, we discuss aspects that strongly influence this impression.

Alignment and Spatial Tracking

Probably the most important aspect influencing usability is the alignment of AR content in relation to the display. While we already talked about the levels of spatial proximity, it is also important to consider the quality of the spatial tracking the AR device provides. *DesignAR's* setup places very high demands on spatial tracking and ideally requires an accuracy in the millimeter range. Although the tracking of the HoloLens is very precise within a room-scale environment, it proved to be problematic for *DesignAR*. Offsets of up to five millimeters are not uncommon, making precise actions like modeling challenging. We mitigated this for *DesignAR* by using a 2D projection of the model on the display (see Figure 4) whenever a modeling action starts, thus re-enabling pixel-perfect interaction. Nonetheless, these tracking imprecisions can be sufficient to break the illusion of using a single seamless system. Besides general offsets, the position of the model also strongly depends on the user's perspective. Whenever it changes significantly, the AR content starts to drift from its supposed position in relation to real world objects. For *DesignAR*, this problem is solved by the mostly static position of the user. However, it should be taken into consideration for other applications, where users for instance move around the interactive surface. Whenever users start the application, AR objects should be at the correct position in relation to the display without the need for adjustment. Therefore, the automatic synchronization between the position of AR content and the display is very important. We tested the world anchors provided by the HoloLens itself, as well as a custom AR-marker based approach using both the *ARToolKit* and *Vuforia* frameworks. None of these solutions offered sufficient precision to eliminate manual adjustment of the content after the initial recognition especially regarding the orientation of the content.

Issues of the HMD's Presentation Quality

The current HoloLens provides a very **limited field-of-view** (FOV) of ca. 30 degrees. *DesignAR* positions users very close to the AR content, leading to frequent clipping issues and requires a lot of head movement. Currently, we found no other solution besides increasing the distance to the display, which in turn interrupts the design process of users. We hope for future AR HMDs to address this limitation.

In contrast to the Surface Studio, the HoloLens offers a significantly **lower pixel resolution**. This has a major impact on text presentation, which needs to be quite large to be comfortably readable in AR. As a consequence, to maintain a consistent size and therefore seamless integration of AR and display, menus in *DesignAR* on the display are considerably larger than would actually be necessary.

The **opacity of the AR content** in relation to the brightness of the Surface Studio, which we anticipated to be a major issue, proved to have little effect in practice. A dark background on the display is still preferable for the better perception of AR content, but brighter objects are nevertheless possible without issues.

Touch Interaction with 3D Objects in Augmented Reality

One of our main goals with *DesignAR* was to use touch interaction for almost all of the features provided. The main reasons for this were the higher precision and the added constraints in comparison to mid-air interaction. While this initially seems like a limitation for interacting with objects in AR, the spatial relationship and proximity of the AR objects to the display counteracts this apparent disadvantage. In our concept, the precision of the input decreased with increasing distance of AR objects to the display, i.e., the proximity levels **L1** to **L3** we described before. For each object situated in **L1**, i.e., directly on or above the display, we suggest interacting directly on the surface of the display using pen input for tasks like modeling or sketching supplemented by touch gestures. For interacting with objects in **L2**, i.e., at the edge of or in close proximity to the display, we propose a frame around the screen's border which is used to manipulate and control the corresponding AR objects (as in the case with *DesignAR*'s orthographic views or the offloaded menus). For every object in **L3**, i.e., objects with no clear spatial relation to the display, we advocate to use mid-air interaction instead of touch (like *DesignAR*'s 3D instances). While the latter would be more precise, it would be too indirect for intuitively manipulating objects beyond the display.

There is a general mismatch in stereoscopic 3D between the **perception of objects** and the 2D interaction of touch and pen input (cf. [22, 10, 51]). While interacting with objects visually located behind the screen (e.g., fish-tank VR) works comparatively well, interacting with objects in front of the screen, like *DesignAR*'s modeled object, is challenging. In this case, users have to reach through the object to touch the interactive surface, which is irritating for many users and makes it difficult to recognize the interaction on the display. First, instead of interacting with the 3D object itself, we tested interacting with the orthographic projection on the display (similar to *Interactive Shadows* by Herndon et al. [26]). However, this severely limits the interaction space, as users could only interact with one side of the model. Furthermore, it also leads to indirect interaction because users look at the 3D model but interact with a proxy weakening the benefits of stereoscopic, three-dimensional representation. Our second approach was to use the screen in a similar way as mirror and render another instance of the modeled object. The instance is displayed in stereoscopic AR as well but translated on the z axis to appear behind the screen instead. This mitigates the problem of the limited interaction space compared to the previous approach, but retains the issues of indirect interaction because users interact with the mirror image instead of the real object. Our final solution was to use the head-coupled perspective projection described before (illustrated in Figure 4). It mitigates the issues of the two previous approaches but temporarily sacrifices the stereoscopic rendering. Nonetheless, feedback by our usability experts was very favorable with regard to this technique.

Responsiveness

Another relevant aspect regarding usability is the perceived responsiveness of the system to user input. Since *DesignAR* consists of two separate applications and most interaction takes place on the display using multi-touch and pen input, every

action that changes AR content must be first relayed to the AR device using the network. We achieved good results with a dedicated wireless network solution and the router being in close proximity to the devices resulting in barely noticeable latency. However, tests using a more saturated network quickly showed that rising latency has an immediate impact on the user experience for all operations requiring continuous updates, like for example moving or rotating the model. Switching from client-server to a peer-to-peer infrastructure would eliminate one layer of indirection and thus reduce latency. However, we conducted no tests in this regard.

ENVISIONING AUGMENTED DISPLAYS

In the previous sections, we have reported on how the combination of head-coupled AR and interactive surfaces can be advantageous for the particular use case of a 3D modeling tool. However, many aspects of *DesignAR* are part of a much larger design space which we call *Augmented Displays*. Our goal is to form the notion of this new class of display systems and the advantageous addition of AR which is not limited to 3D objects but also enables to overlay 2D content on the display in a seamless manner. The AR content can be situated directly on the display, next to it, or make use of the whole environment that surrounds the user. Realizing the potential of a HMD, even personalized views in multi-user scenarios are possible. There is existing work which can be considered as *Augmented Displays* from De Araujo et al. [15], Butscher et al. [12], and Riemann et al. [46]. They all combine interactive tabletops with some form of Augmented Reality overlay for use cases like modeling or visualization. However, other domains also promise interesting applications especially for data that is intrinsically three-dimensional, such as architecture or medicine.

After giving a definition of *Augmented Displays*, we want to briefly illustrate parts of its design space on the example of roles of AR objects (hereafter called augmentations) and their spatial alignment with the display.

*We define the concept of **Augmented Displays** as the extension of an non-stereoscopic, interactive surface (e.g., tablet, tabletop, display wall), with two- or three-dimensional content using personal Augmented Reality devices. The display serves as a frame of reference for all associated augmentations. **Augmented Displays** are situated in between Augmented Reality and Augmented Virtuality (cf. Mixed Reality continuum [40]) and represent a subgroup of distributed user interfaces.*

We distinguish the role of an augmentation to be either primary, which places the focus of the augmentation and confines the display in a supplementary role (see *DesignAR* modeled Objects, Figure 1a) or to be secondary where augmentations complement specific content on the display (see *DesignAR*'s objects browser, Figure 1b). We previously highlighted the importance of proximity between Augmentations and the display. Another important aspect in this regard is the alignment of augmentations with either the display itself, like *DesignAR*'s orthographic views (see Figure 1a), or specific objects on the display, like *DesignAR*'s offloaded menu items (see Figure 6). Consequently, augmentations should also react to changes on

the display and be updated accordingly when, e.g., an object on the display was moved or has changed its state.

We are excited to further explore our vision of *Augmented Displays* in future work. Our goal is not only to find additional use cases but also to define a comprehensive design space, to further classify existing work within, and to explore different types of device setups. More precisely, we want to investigate which other roles AR objects can have in relation to objects on the display or the display itself. Another research question is which zones can be defined in relation to the form factor of the display, and how AR content interacts with them, e.g., by being clipped. The alignment and the spatial link between AR and display content is also of interest as well as the display problems of 3D content compared to 2D content. While constraint interaction has been proven to be successful by prior research, an important research question is how the naturalness and familiarity of touch interaction translates to interacting with 3D AR objects and how this compares to interacting in purely virtual environments.

CONCLUSION

We presented *DesignAR*, combining an interactive surface with multi-touch and pen input and a head-mounted, stereoscopic Augmented Reality display to create an augmented design workstation for constructing simple 3D models. We proposed several concepts for supporting the user's design process, utilizing the AR space above and beyond the display with techniques like AR orthographic views, UI offloading and interaction, as well as freely placeable model instances for real world reference. Furthermore, we presented our prototype and reported on challenges faced and insights gained. While we are completely aware that our implementation is far from being a fully fledged 3D modeling application, we hope that we could demonstrate the power of this novel display configuration and our devised concepts in this domain. We have also discussed that the combination of AR and interactive surfaces constitutes a new class of display systems which we coined *Augmented Displays*.

In the future, we plan to further improve and expand our prototype and evaluate our techniques in a formative user study with experts in the fields of modeling and 3D design. We see *DesignAR* as a promising stepping stone on the path to *Augmented Displays* and we plan to further expand our vision on this exciting new class of devices that take advantage of both the natural input and haptics of interactive surfaces and the imaginative power of immersive augmentations.

ACKNOWLEDGEMENTS

This work was partially funded by DFG grant 389792660 as part of TRR 248 (see <https://perspicuous-computing.science>). We would like to thank Severin Engert for his support in creating the figures and videos for this publication.

REFERENCES

- [1] 2015. Microsoft HoloLens: Partner Demo with Maya by Autodesk. <https://youtu.be/pdh-9p01wpQ>. (2015). Accessed: 2019-06-28.
- [2] 2018. TiltBrush. <https://www.tiltbrush.com/>. (2018). Accessed: 2018-04-27.
- [3] 2019. grib. <https://grib3d.com/>. (2019). Accessed: 2019-28-06.
- [4] 2019. Shapr3D. <https://www.shapr3d.com>. (2019). Accessed: 2019-06-28.
- [5] 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 (CHI '18)*. ACM, New York, NY, USA, Article 185, 15 pages. DOI: <http://dx.doi.org/10.1145/3173574.3173759>
- [6] 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, New York, NY, USA, 151–160. DOI: <http://dx.doi.org/10.1145/1449715.1449740>
- [7] Tim Bakker, Jouke Verlinden, David Abbink, and Roel van Deventer. 2017. Development of a Haptic Device with Tactile and Proprioceptive Feedback for Spatial Design Tasks. In *2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*. 223–228. DOI: <http://dx.doi.org/10.1109/ISMAR-Adjunct.2017.74>
- [8] Hrvoje Benko, Andy Wilson, and Federico Zannier. 2014. Dyadic Projected Spatial Augmented Reality. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, 645–655. <https://www.microsoft.com/en-us/research/publication/dyadic-projected-spatial-augmented-reality/>
- [9] Natalia Bogdan, Tovi Grossman, and George Fitzmaurice. 2014. HybridSpace: Integrating 3D freehand input and stereo viewing into traditional desktop applications. In *2014 IEEE Symposium on 3D User Interfaces (3DUI)*. 51–58. DOI: <http://dx.doi.org/10.1109/3DUI.2014.6798842>
- [10] Gerd Bruder, Frank Steinicke, and Wolfgang Stuerzlinger. 2013. Touching the Void Revisited: Analyses of Touch Behavior on and above Tabletop Surfaces. In *Human-Computer Interaction – INTERACT 2013*, Paula Kotzé, Gary Marsden, Gitte Lindgaard, Janet Wesson, and Marco Winckler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 278–296.
- [11] Rahul Budhiraja, Gun A. Lee, and Mark Billinghurst. 2013. Interaction Techniques for HMD-HHD Hybrid AR Systems. In *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 243–244. DOI: <http://dx.doi.org/10.1109/ISMAR.2013.6671786>
- [12] Simon Butscher, Sebastian Hubenschmid, Jens Müller, Johannes Fuchs, and Harald Reiterer. 2018. Clusters, Trends, and Outliers: How Immersive Technologies Can Facilitate the Collaborative Analysis of Multidimensional Data. In *Proceedings of the 2018 CHI*

- Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 90, 12 pages. DOI:<http://dx.doi.org/10.1145/3173574.3173664>
- [13] Jeff Butterworth, Andrew Davidson, Stephen Hench, and Marc. T. Olano. 1992. 3DM: A Three Dimensional Modeler Using a Head-mounted Display. In *Proceedings of the 1992 Symposium on Interactive 3D Graphics (I3D '92)*. ACM, New York, NY, USA, 135–138. DOI:
<http://dx.doi.org/10.1145/147156.147182>
- [14] Andreas Butz, Tobias Höllerer, Steven Feiner, Blair MacIntyre, and Clifford Beshers. 1999. Enveloping Users and Computers in a Collaborative 3D Augmented Reality. In *Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR '99)*. IEEE Computer Society, Washington, DC, USA, 35–45.
<http://dl.acm.org/citation.cfm?id=857202.858143>
- [15] Bruno R. De Araujo, 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>
- [16] Fabrice Declec, Martin Hachety, and Pascal Guitton. 2009. Tech-note - ScrutiCam: Camera Manipulation Technique for 3D Objects Inspection. In *2009 IEEE Symposium on 3D User Interfaces*. 19–22. DOI:
<http://dx.doi.org/10.1109/3DUI.2009.4811200>
- [17] 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>
- [18] John J. Dudley, Hendrik Schuff, and Per Ola Kristensson. 2018. Bare-Handed 3D Drawing in Augmented Reality. In *Proceedings of the 2018 Designing Interactive Systems Conference (DIS '18)*. ACM, New York, NY, USA, 241–252. DOI:
<http://dx.doi.org/10.1145/3196709.3196737>
- [19] Niklas Elmqvist. 2011. *Distributed User Interfaces: State of the Art*. Springer London, London, 1–12. DOI:
http://dx.doi.org/10.1007/978-1-4471-2271-5_1
- [20] Steven Feiner and Ari Shamash. 1991. Hybrid User Interfaces: Breeding Virtually Bigger Interfaces for Physically Smaller Computers. In *Proceedings of the 4th Annual ACM Symposium on User Interface Software and Technology (UIST '91)*. ACM, New York, NY, USA, 9–17. DOI:
<http://dx.doi.org/10.1145/120782.120783>
- [21] Mathias Frisch, Ricardo Langner, and Raimund Dachselt. 2011. NEAT: A Set of Flexible Tools and Gestures for Layout Tasks on Interactive Displays. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces 2011 (ITS '11)*. ACM, New York, NY, USA, 1–10. DOI:
<http://dx.doi.org/10.1145/2076354.2076356>
- [22] Tovi Grossman and Daniel Wigdor. 2007. Going Deeper: a Taxonomy of 3D on the Tabletop. In *Second Annual IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP'07)*. 137–144. DOI:
<http://dx.doi.org/10.1109/TABLETOP.2007.18>
- [23] Jens Grubert, Michel Pahud, Raphael Grasset, Dieter Schmalstieg, and Hartmut Seichter. 2015. The Utility of Magic Lens Interfaces on Handheld Devices for Touristic Map Navigation. *Pervasive Mob. Comput.* 18, C (April 2015), 88–103. DOI:
<http://dx.doi.org/10.1016/j.pmcj.2014.08.005>
- [24] Martin Hachet, Benoit Bossavit, Aurélie Cohé, and Jean-Baptiste de la Rivière. 2011. Toucheo: Multitouch and Stereo Combined in a Seamless Workspace. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, New York, NY, USA, 587–592. DOI:
<http://dx.doi.org/10.1145/2047196.2047273>
- [25] Mark Hancock, Thomas ten Cate, and Sheelagh Cappendale. 2009. Sticky Tools: Full 6DOF Force-based Interaction for Multi-touch Tables. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09)*. ACM, New York, NY, USA, 133–140. DOI:
<http://dx.doi.org/10.1145/1731903.1731930>
- [26] Kenneth P. Herndon, Robert C. Zeleznik, Daniel C. Robbins, D. Brookshire Conner, Scott S. Snibbe, and Andries van Dam. 1992. Interactive Shadows. In *Proceedings of the 5th Annual ACM Symposium on User Interface Software and Technology (UIST '92)*. ACM, New York, NY, USA, 1–6. DOI:
<http://dx.doi.org/10.1145/142621.142622>
- [27] Ken Hinckley, Koji Yatani, Michel Pahud, Nicole Coddington, Jenny Rodenhouse, Andy Wilson, Hrvoje Benko, and Bill Buxton. 2010. Pen + Touch = New Tools. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology (UIST '10)*. ACM, New York, NY, USA, 27–36. DOI:
<http://dx.doi.org/10.1145/1866029.1866036>
- [28] 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/Addison-Wesley Publishing Co., New York, NY, USA, 409–416. DOI:
<http://dx.doi.org/10.1145/311535.311602>
- [29] 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. DOI:
<http://dx.doi.org/10.1109/TVCG.2016.2518099>
- [30] Sung-A Jang, Graham Wakefield, and Sung-Hee Lee. 2017. Incorporating Kinesthetic Creativity and Gestural Play into Immersive Modeling. In *Proceedings of the 4th International Conference on Movement Computing*

- (MOCO '17). ACM, New York, NY, USA, Article 17, 8 pages. DOI : <http://dx.doi.org/10.1145/3077981.3078045>
- [31] Brett R. Jones, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. 2013. IllumiRoom: Peripheral Projected Illusions for Interactive Experiences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 869–878. DOI : <http://dx.doi.org/10.1145/2470654.2466112>
- [32] 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 (Sep. 2007), 1067–1081. DOI : <http://dx.doi.org/10.1109/TVCG.2007.1060>
- [33] Yongkwan Kim, Sang-Gyun An, Joon Hyub Lee, and Seok-Hyung Bae. 2018. Agile 3D Sketching with Air Scaffolding. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 238, 12 pages. DOI : <http://dx.doi.org/10.1145/3173574.3173812>
- [34] 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>
- [35] Jinha Lee, Alex Olwal, Hiroshi Ishii, and Cati Boulanger. 2013. SpaceTop: Integrating 2D and Spatial 3D Interactions in a See-through Desktop Environment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 189–192. DOI : <http://dx.doi.org/10.1145/2470654.2470680>
- [36] Zhen Li, Michelle Annett, Ken Hinckley, Karan Singh, and Daniel Wigdor. 2019. HoloDoc: Enabling Mixed Reality Workspaces That Harness Physical and Digital Content. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, Article 687, 14 pages. DOI : <http://dx.doi.org/10.1145/3290605.3300917>
- [37] Tim McGraw, Esteban Garcia, and Drew Sumner. 2017. Interactive Swept Surface Modeling in Virtual Reality with Motion-tracked Controllers. In *Proceedings of the Symposium on Sketch-Based Interfaces and Modeling (SBIM '17)*. ACM, New York, NY, USA, Article 4, 9 pages. DOI : <http://dx.doi.org/10.1145/3092907.3092908>
- [38] Daniel Mendes, Daniel Medeiros, Mauricio Sousa, Ricardo Ferreira, Alberto Raposo, Alfredo Ferreira, and Joaquim Jorge. 2017. Mid-air Modeling with Boolean Operations in VR. In *2017 IEEE Symposium on 3D User Interfaces (3DUI)*. 154–157. DOI : <http://dx.doi.org/10.1109/3DUI.2017.7893332>
- [39] Leonel Merino, Mario Hess, Alexandre Bergel, Oscar Nierstrasz, and Daniel Weiskopf. 2019. PerfVis: Pervasive Visualization in Immersive Augmented Reality for Performance Awareness. In *Companion of the 2019 ACM/SPEC International Conference on Performance Engineering (ICPE '19)*. ACM, New York, NY, USA, 13–16. DOI : <http://dx.doi.org/10.1145/3302541.3313104>
- [40] Paul Milgram, Haruo Takemura, Akira Utsumi, and Fumio Kishino. 1995. Augmented Reality: A Class of Displays on the Reality-Virtuality Continuum. In *Telemanipulator and Telepresence Technologies (Proceedings of the SPIE)*, H. Das (Ed.), Vol. 2351. 282–292. DOI : <http://dx.doi.org/10.1117/12.197321>
- [41] Alexandre Millette and Micheal J. McGuffin. 2016. DualCAD: Integrating Augmented Reality with a Desktop GUI and Smartphone Interaction. In *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*. 21–26. DOI : <http://dx.doi.org/10.1109/ISMAR-Adjunct.2016.0030>
- [42] Mark Mine, Arun Yoganandan, and Dane Coffey. 2014. Making VR Work: Building a Real-world Immersive Modeling Application in the Virtual World. In *Proceedings of the 2Nd ACM Symposium on Spatial User Interaction (SUI '14)*. ACM, New York, NY, USA, 80–89. DOI : <http://dx.doi.org/10.1145/2659766.2659780>
- [43] Patrick Paczkowski, Julie Dorsey, Holly Rushmeier, and Min H. Kim. 2014. Paper3D: Bringing Casual 3D Modeling to a Multi-touch Interface. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, NY, USA, 23–32. DOI : <http://dx.doi.org/10.1145/2642918.2647416>
- [44] Wayne Piekarski. 2006. 3D Modeling with the Tinmith Mobile Outdoor Augmented Reality System. *IEEE Computer Graphics and Applications* 26, 1 (Jan 2006), 14–17. DOI : <http://dx.doi.org/10.1109/MCG.2006.3>
- [45] Jun Rekimoto and Masanori Saitoh. 1999. Augmented Surfaces: A Spatially Continuous Work Space for Hybrid Computing Environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*. ACM, New York, NY, USA, 378–385. DOI : <http://dx.doi.org/10.1145/302979.303113>
- [46] Jan Riemann, Markus Funk, Martin Schmitz, Christian Meurisch, and Max Mühlhäuser. 2018. OverTop: Breaking the Boundaries of Tangible Tabletop Environments. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers (UbiComp '18)*. ACM, New York, NY, USA, 231–234. DOI : <http://dx.doi.org/10.1145/3267305.3267559>
- [47] Naoki Sasaki, Hsiang-Ting Chen, Daisuke Sakamoto, and Takeo Igarashi. 2013. Facetons: Face Primitives with Adaptive Bounds for Building 3D Architectural

Models in Virtual Environment. In *Proceedings of the 19th ACM Symposium on Virtual Reality Software and Technology (VRST '13)*. ACM, New York, NY, USA, 77–82. DOI : <http://dx.doi.org/10.1145/2503713.2503718>

- [48] Mickael Sereno, Lonni Besançon, and Tobias Isenberg. 2019. Supporting Volumetric Data Visualization and Analysis by Combining Augmented Reality Visuals with Multi-Touch Input. In *Posters at the Eurographics/IEEE VGTC Annual Visualization Conference (EuroVis)*.
- [49] Jeff K. T. Tang, Tin-Yung Au Duong, Yui-Wang Ng, and Hoi-Kit Luk. 2015. Learning to Create 3D Models via an Augmented Reality Smartphone Interface. In *2015 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*. 236–241. DOI : <http://dx.doi.org/10.1109/TALE.2015.7386050>
- [50] James Tompkin, Samuel Muff, James McCann, Hanspeter Pfister, Jan Kautz, Marc Alexa, and Wojciech Matusik. 2015. Joint 5D Pen Input for Light Field Displays. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & #38; Technology (UIST '15)*. ACM, New York, NY, USA, 637–647. DOI : <http://dx.doi.org/10.1145/2807442.2807477>
- [51] Dimitar Valkov, Frank Steinicke, Gerd Bruder, and Klaus Hinrichs. 2011. 2D Touching of 3D Stereoscopic Objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 1353–1362. DOI : <http://dx.doi.org/10.1145/1978942.1979142>
- [52] Colin Ware, Kevin Arthur, and Kellogg S. Booth. 1993. Fish Tank Virtual Reality. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (CHI '93)*. ACM, New York, NY, USA, 37–42. DOI : <http://dx.doi.org/10.1145/169059.169066>
- [53] 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, New York, NY, USA, 3855–3864. DOI : <http://dx.doi.org/10.1145/2556288.2557090>
- [54] 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, New York, NY, USA, 167–174. DOI : <http://dx.doi.org/10.1145/505008.505041>
- [55] 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, New York, NY, USA, 223–226. DOI : <http://dx.doi.org/10.1145/1450579.1450627>
- [56] Lingyun Yu, Pjotr Svetachov, Petra Isenberg, Maarten Everts, and Tobias Isenberg. 2010. FI3D: Direct-Touch Interaction for the Exploration of 3D Scientific Visualization Spaces. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (2010), 1613–1622. DOI : <http://dx.doi.org/10.1109/TVCG.2010.157>
- [57] Xiaoqiang Zhu, Lei Song, Lihua You, Mengyao Zhu, Xiangyang Wang, and Xiaogang Jin. 2017. Brush2Model: Convolution Surface-Based Brushes for 3D Modelling in Head-Mounted Display-Based Virtual Environments. *Computer Animation and Virtual Worlds* 28, 3-4 (2017), e1764. DOI : <http://dx.doi.org/10.1002/cav.1764>
- [58] Sisi Zlatanova. 2001. 3D Modelling for Augmented Reality. *International Archives of Photogrammetry and Remote Sensing, Volume XXXIV, Part 2W2, The 3rd ISPRS Workshop on Dynamic and Multi- Dimensional GIS and The 10th Annual Conference of CPGIS on Geoinformatics, Bangkok, 2001* (01 2001).