Portalware: Exploring Free-Hand AR Drawing with a Dual-Display Smartphone-Wearable Paradigm

Jing Qian*
jing_qian@brown.edu
Brown University
Providence RI, USA

Tongyu Zhou*
tongyu_zhou@brown.edu
Brown University
Providence RI, USA

Meredith Young-Ng*
meredith_young-ng@brown.edu
Brown University
Providence RI, USA

Jiaju Ma
jiaju_ma@brown.edu
Brown University
Providence RI, USA

Angel Cheung
angel_cheung@brown.edu
Brown University
Providence RI, USA

Xiangyu Li
xiangyu_li@brown.edu
Brown University
Providence RI, USA

Ian Gonsher
ian_gonsher@brown.edu
Brown University
Providence RI, USA

Jeff Huang
jeff_huang@brown.edu
Brown University
Providence RI, USA

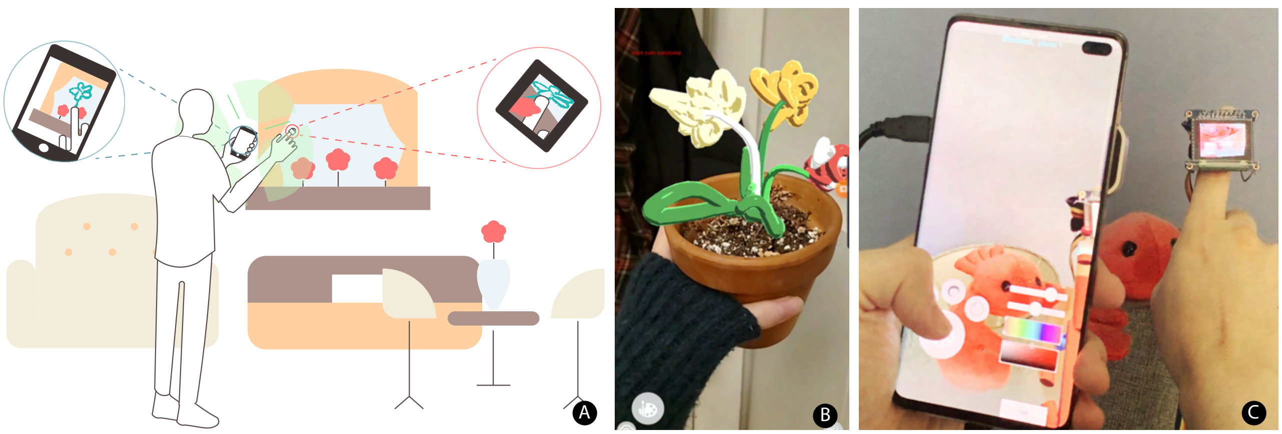


Figure 1: Portalware features a mobile setup for dual-display representation of augmented reality (AR) objects viewed through a smartphone and a wearable. A) Illustration of a user sketching AR content on both the smartphone and the wearable. B) An AR sketch of virtual flowers on a physical pot with Portalware. C) The user's view of Portalware.

ABSTRACT

Free-hand interaction enables users to directly create artistic augmented reality content using a smartphone, but lacks natural spatial depth information due to the small 2D display's limited visual feedback. Through an autobiographical design process, three authors explored free-hand drawing over a total of 14 weeks. During this process, they expanded the design space from a single-display smartphone format to a dual-display smartphone-wearable format (Portalware). This new configuration extends the virtual content from a smartphone to a wearable display and enables multi-display free-hand interactions. The authors documented experiences where 1) the display extends the smartphone's canvas perceptually, allowing the authors to work beyond the smartphone screen view; 2) the additional perspective mitigates the difficulties of depth perception and improves the usability of direct free-hand manipulation; 3) the wearable use cases depend on the nature of the drawing, such as: replicating physical objects, "in-situ" mixed reality pieces, and multi-planar drawings.

CCS CONCEPTS

• **Human-centered computing → Mixed / augmented reality**; *Participatory design*; *Mobile phones*.

*The first three authors contributed similar effort.

DIS '21, June 28-July 2, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-8476-6/21/06…$15.00
https://doi.org/10.1145/3461778.3462098

KEYWORDS

wearable, free-hand drawing, autobiographical design, augmented reality

ACM Reference Format:

Jing Qian, Tongyu Zhou, Meredith Young-Ng, Jiaju Ma, Angel Cheung, Xiangyu Li, Ian Gonsher, and Jeff Huang. 2021. Portalware: Exploring Free-Hand AR Drawing with a Dual-Display Smartphone-Wearable Paradigm. In *Designing Interactive Systems Conference 2021 (DIS '21), June 28-July 2, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3461778.3462098

1 INTRODUCTION

Since the earliest cave paintings, drawing has taken the 3-dimensional world and flattened it onto a 2-dimensional plane. With traditional media, sketching in 3 dimensions is impossible as ink cannot exist in air and is constrained to a flat surface. However, in augmented reality (AR), we move one step closer to truly 3-dimensional sketching, as AR displays are capable of overlaying virtual ink onto a physical environment, allowing sketching to occur *nowhere and everywhere* simultaneously. By allowing users to augment tangible objects, AR sketching enables new scenarios such as pictorially communicating house sitting instructions, annotating landmarks in public spaces, and inspiring individuals to re-imagine once familiar places.

Head-mounted displays (HMDs) provide excellent accuracy and immersion for sketching in AR, but lack ubiquitous mobility, making them difficult to support sketching in different everyday scenarios. Given that smartphones and wearable devices are lightweight, ubiquitous, and generally accessible, they have the potential to support a more contextual and mobile AR sketching experience. While smartphones are powerful AR devices, smaller, watch-like wearables are rarely used on their own in AR applications due to their limited display size and usual lack of an on-board camera. Prior work explore a smartphone-wearable combination, using the wearable to support additional sensing [52] or input [21, 59] for AR. However, little is known about what effect this combination has for free-hand AR sketching and what benefits and challenges the wearable brings. Given that these lightweight devices are promising candidates for mobile AR sketching, what interaction paradigms need to be explored?

Smartphone AR interactions typically occur via 2D screen input: users touch their screen to interact with 3D objects visible from their viewing location and perspective. However, because the input is 2D, additional transformations are required to achieve a fully 3D image, such as 2D proxy plans that align strokes made from 2D input into 3D space [37]. Unlike 2D screen input, 3D interactions such as free-hand gesture or device input directly generate strokes in 3D space (e.g., Google's Tilt Brush [26]). Despite being easy-to-use, immersive, and fun, free-hand 3D sketching is not available on most smartphones due to hardware and tracking limitations; thus, the exact experience and design lessons for smartphones are unknown.

In this paper, we explore factors that may shape the experience of portable 3D free-hand sketching in AR. Our design process builds on an open-source system (*Portal-ble*) with a dual-fisheye depth camera for a larger free-hand interaction region [54]. To better understand long-term usage, interaction challenges for free-hand AR sketching, and adhere to COVID-19 restrictions, we followed an autobiographical design procedure [47] to iteratively experience, test, and improve the sketching prototype over a total of 14 weeks. Three authors with design and computing backgrounds documented their experience using the prototype for their own drawings (e.g., Figure 1B).

The three authors started by using the single-display smartphone prototype to create in-situ annotations, markers, and paintings. They identified personal pain points and design lessons, including visual perspective (e.g., stroke alignment), ergonomics (e.g., minimizing strain), and hand-eye coordination. The authors noted that a visual depth indicator that highlights where the finger intersects the stroke, load and save functions, and a spatial 3D user interface (UI) helped mitigate these issues. This exploration also revealed that the smartphone's limited screen size could only show a small drawing canvas, requiring them to follow the gesturing hand by moving the smartphone to keep the sketch in view. This prevented them from interacting with the region around the smartphone or using the user's peripheral vision, helpful affordances identified in prior work [42, 58].

Further explorations by the same three authors focused on whether dual-display AR sketching, which we refer to as *Portalware* (Figure 1A), helps mitigate these limitations. By incorporating an additional wearable mounted to the fingertip (Figure 1C) to explore dual-display mobile sketching, they aimed to make better use of peripheral vision and interaction regions. They found that the wearable display *extends the perceptual canvas* and helps visualize content in the interaction region around the mobile device. This visual feedback can be used either for triggering 3D interactions or as a user interface for a fluid sketch workflow. The authors also found that glimpsing and switching their focus between the two screens help improve the overall sketching experience. For detailed design lessons, refer to Sections 4.3.4 and 4.4.4.

Our contributions are: 1) a description of the Portalware system, which enables free-hand AR sketching for both a single-display and dual-display format; and 2) design lessons gleaned from autobiographical design reports that explore perceptual, workflow, ergonomic, and interaction challenges for free-hand sketching for both single-display and dual-display formats.

2 RELATED WORK

2.1 Sketching in AR and VR

Sketching systems in AR and VR use a variety of mid-air hand gestures as 3D input, spanning from pens and controllers to free-hand gestures. Pens, a common sketching apparatus, have frequently been used for sketching in AR and VR in mid-air or on a 2D tablet [3, 23, 30, 38, 39, 62]. 3D pen-based systems, such as ARPen and VRSketchIn, extend pen-based sketching into 3D space [18, 67]. In VR, handheld controllers with an HMD are also often used for sketching, as they are intuitive and provide haptic feedback [24, 26, 40, 44]; however, this still poses challenges as users are not accustomed to sketching without a hard surface and need additional visual cues to sketch accurately [4]. In AR, 3D sketching is challenging not only due to visual perception issues in VR, but also due to possible interactions with physical objects [36]. Without VR's reliance on HMDs, AR controllers have expanded to other forms to remedy this issue, such as reusing a tablet's motion trajectory [39]. Similarly to VR, additional affordances and visual guidelines can help improve accuracy and task completion, particularly when sketching next to physical objects [37, 68]. Ultimately, directly sketching in 3D enables users to sketch freely without having

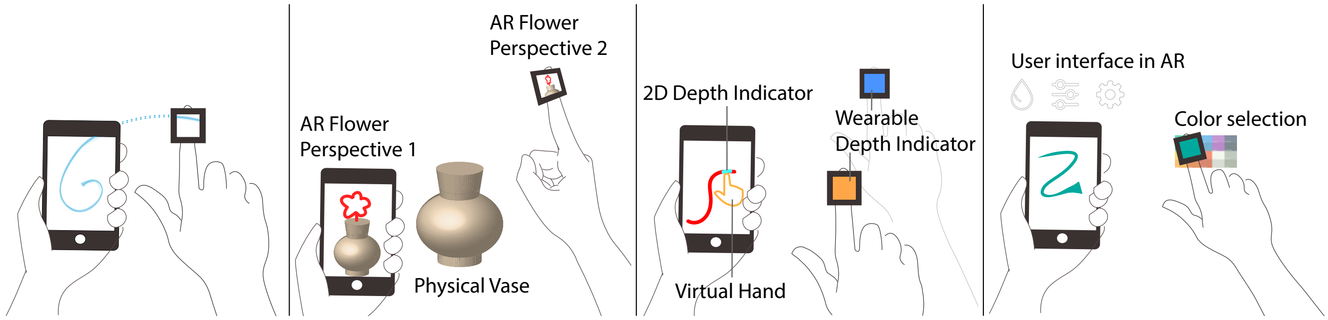


Figure 2: Examples of smartphone-wearable interactions enabled by Portalware. *Left*: The inclusion of the wearable expands the region of visible AR content. *Middle-Left*: The wearable allows the user to observe AR objects from different perspectives from the smartphone. *Middle-Right*: The wearable display can indicate physical depth by changing its background color to its distance to the smartphone. *Right*: The AR space near the smartphone screen can be accessed by the wearable, allowing a larger working space.

to generate 3D content from 2D input; however, this can present alignment and accuracy difficulties with 3D strokes and planes. Corrective measures such as snapping can help users align 3D strokes [7, 37]. While these corrective post-processing measures can help improve alignment accuracy, they fail to improve the artistic workflow itself.

Uninstrumented (bare) free-hand manipulation is often considered an intuitive interaction for 3D objects [8, 15, 31, 54, 75] and navigating bi-manual interaction menus [58], yet has not been thoroughly explored for 3D sketching in AR or VR. While free-hand sketching has excelled in tasks such as writing characters on tablets and computer screens [66], this mode of interaction remains challenging for surfaceless situations [63]. Current work in free-hand sketching explores several applications: spatial annotations [10], modeling [43], and rapid prototyping [32]. However, their performance is still largely affected by user technique. For example, fast free-form sketching is still not as accurate as tapline techniques, in which stroke lines are managed by control points [20]. Similarly, a lack of eye-hand coordination and haptic feedback has also been shown to be detrimental to free-hand sketching [32].

In addition to improving the experience of free-hand AR drawing, we are interested in improving overall sketch quality. Previous studies note that 3D free-hand drawing quality naturally improves over time with practice [71]. However, the addition of physical and visual guides can further help users position strokes more accurately, with the caveat that these guides may negatively affect stroke aesthetics [4]. An alternative way to convey physical guidance is via a haptic-based design interface, shown to improve user cognitive abilities and engagement without disrupting the stroke itself [55].

Compared to previous AR and VR sketching systems, our work focuses on exploring interaction affordances for free-hand sketching in smartphone AR. We build upon existing design lessons and interaction affordances from 3D sketching to help users better adapt to making high-quality 3D sketches in smartphone AR.

2.2 Wearable Displays

Existing smartphone-wearable AR devices use a smartphone as the primary device with a secondary wearable display or haptic device, such as a paired AR smartphone and wristband display with wearable haptics for sensing the weight of virtual objects [52]. Most smartphone-wearable AR dual-display devices, however, repurpose existing smartwatches as secondary displays. Studies have examined the foreground/background interactions between the two devices [11], with applications ranging from sharing skiing conditions on ski resort maps [21] to visual search [59]. These wearables function as additional input devices to supplement the smartphone, failing to capitalize on their output display capabilities to visualize AR objects within the user's current environment.

Alternative locations for hand-mounted wearables other than the wrist include the fingernail and around the finger (like a ring). The majority of existing finger and fingernail-mounted devices embed different types of sensors such as a RFID tag [65, 70], a small haptic motor [1, 2, 22, 49], a capacitive touch controller [34], and a camera [22, 45] to the top side of the finger or fingernail for additional sensing. These input sensors empower finger-focused interactions ranging from typing [35, 48] to gesture input [28, 76]. In contrast, finger-mounted sensors such as NotiRing are used as interactive wearable display outputs for tactile notifications [56]. Most similar to our work is the usage of a single small OLED display attached to the top of the fingernail [61, 73]; however, both focus on exploring the applications of finger-focused input interactions instead of AR content. In AR, fingernail and finger-mounted input sensors are used to supplement HMD content in FingerTouch, a fingernail-mounted inertial measurement unit sensor paired with an AR HMD [51], and in a finger-mounted camera to magnify content paired with a Hololens [60].

Wearable peripheral displays use persistence of vision and AR HMD extensions to expand their limited field-of-view and enable peripheral object perception. These persistence of vision devices utilize spatial memory [41] and projected image annotations [57] to publicly share content. In contrast, HMDs enable more privacy when sharing information, such as in HandshakAR, which pairs either an existing smartwatch or wrist-mounted smartphone with a Google Glass HMD [6]. In addition, these HMDs can be augmented with sparse peripheral LED displays to expand field-of-view [27, 74].

In contrast to other smartphone-wearable AR systems, our wearable expands the smartphone AR free-hand interaction region by acting as a secondary AR display, sharing the same AR scene with

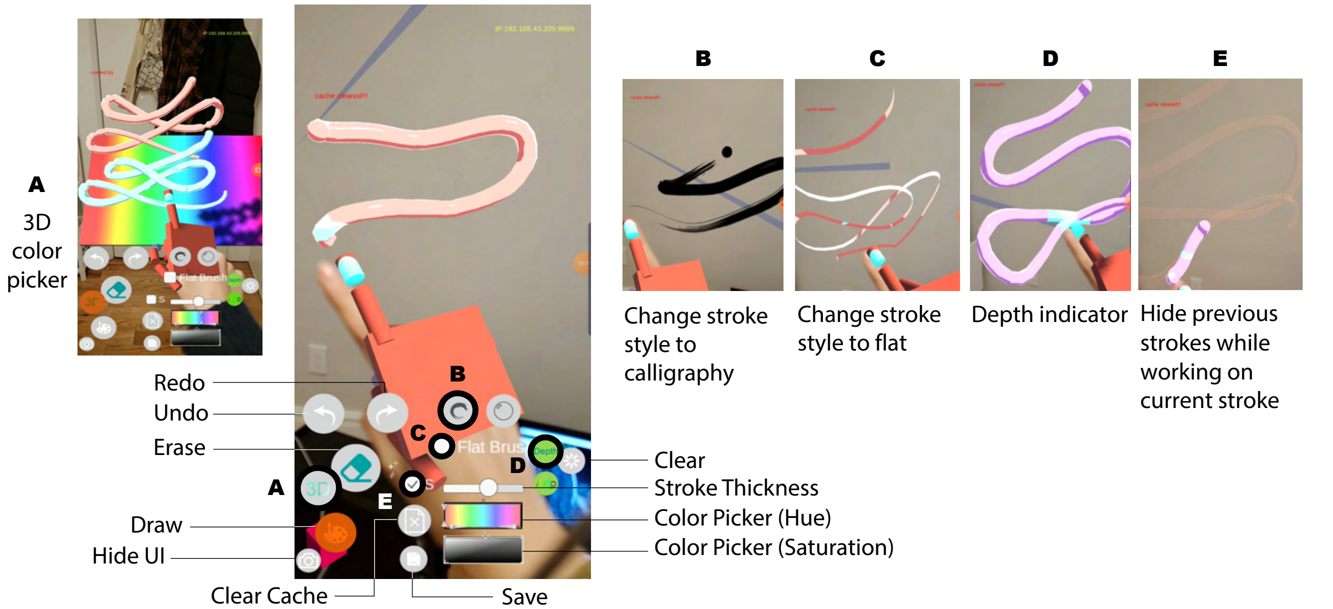


Figure 3: The Portalware smartphone user interface, which includes the following functions: draw, erase, undo, redo, hide UI, save cache, clear cache, clear, change thickness, change hue (2D), change hue (3D), change saturation, change stroke style (calligraphy, flat), toggle depth indicator, and hide previous strokes.

the smartphone in real-time. This secondary display format supports further immersion through additional visual feedback, spatial awareness, and precision when interacting with AR content beyond the smartphone's view.

2.3 Depth Perception

Existing studies focus on analyzing the depth perception problem in mobile AR [25]. Users tend to underestimate distances, as they primarily rely on the height of the visual field as a depth cue [17]. While AR X-ray visualization techniques seem to not affect users' depth perception [17], more effective rendering approaches to address depth perception include device and user-perspective rendering in smartphone AR [13] and scene warping in VR HMDs [53]. Other approaches in smartphone AR focus on generating and processing depth maps for more realistic, geometry-aware 3D interactions [19, 64]. Different visualizations to address perceptual issues with a mid-air pointing device and smartphone AR setup have also been compared [69]. We focus specifically on exploring free-hand sketching on a smartphone and a smartphone-wearable system with depth indicator techniques.

3 PORTALWARE SYSTEM

Inspired by challenges in perception, mobility and interaction from prior work, we aim to build a system that supports free-hand sketching exploration on smartphones. This system should enable users to sketch, modify, and stylize 3D strokes with mid-air hand gestures. Furthermore, we want to explore whether there are genuine uses for a smartphone-wearable sketching system. Thus, the Portalware system features both *a stand-alone smartphone mode* and *a paired smartphone and wearable mode for dual-display shared AR interactions*. During the system iterations, authors engaged in autobiographical design protocols (Section 4) to provide feedback.

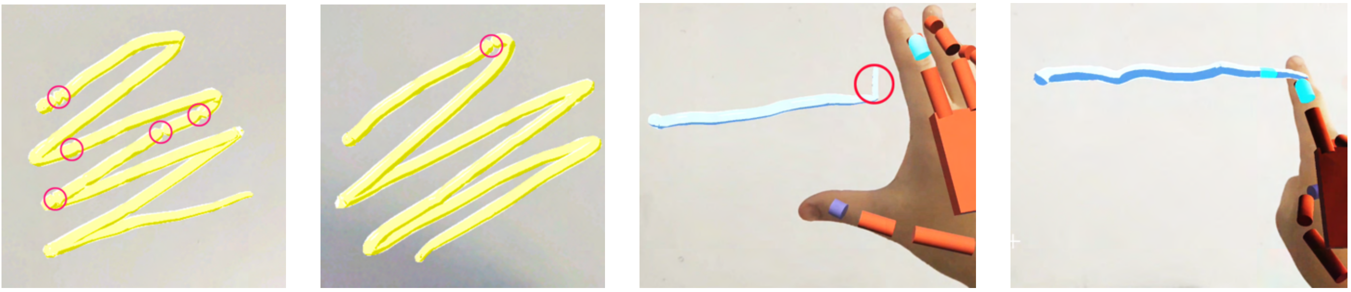
3.1 Smartphone Interface

Portalware runs on smartphones with Android 7.0+ that support ARCore. Sketching interactions, drawing tool commands, and AR rendering are implemented in Unity. A breakdown of the user interface components for Portalware's smartphone application is shown in Figure 3, showing basic sketching functionality: draw, erase, undo, redo, clear, save, clear save, change stroke color, and change stroke thickness. A user draws by holding down the draw button with one hand and using mid-air gestures with the other. Other sketching functions are invoked through button presses as described below.

3.1.1 3D Color Picker. The color picker can be projected onto 3D space by pressing and holding a corresponding button on the left-hand side of the interface (see **A** in Figure 3). While the color picker is displayed, the user can move their drawing hand horizontally to select a hue. Upon button release, this hue is retained for the next stroke.

3.1.2 Depth Indicator. This button serves as a toggle for an indicator that depicts how close the drawing hand is to a virtual object. When the user hovers their index fingertip close to an existing stroke, the nearby portion of the stroke is highlighted light blue to provide a visual alternative to touch (see **D** in Figure 3).

3.1.3 Hide Previous Strokes. When this toggle button is on, all previous strokes are hidden (lowered opacity) when the draw button is held down and the user is working on an existing stroke (see **E** in Figure 3), allowing the user to create strokes that otherwise would



(a) A constant sampling rate generates uneven overlaps (red circles) when the hand moves too slowly.

(b) Adaptive sampling rates dynamically change the sampling interval to avoid overlaps (red circles).

(c) Using the pinch gesture to sketch has a clean beginning, but leaves a tiny tail (red circles) when the gesture changes from pinch to release or idle.

(d) A multimodal interaction that combines a button press with the finger pointing gesture leaves a clean beginning and ending, mitigating the "live mic" issue [72].

Figure 4: A wearable screen on the index finger enables a dual-display multimodal sketching experience in mobile AR.

be occluded by existing ones. When the draw button is released, all strokes become visible again.

3.2 Free-hand Sketching on Smartphones

Informed by prior work, we focus on ensuring that the strokes appear smooth and *high quality*, developing a *depth indicator* to help improve stroke alignment, and empowering *free-hand sketching gesture interactions*.

3.2.1 Stroke Quality. Inspired by SymbiosisSketch [3], we use time and movement speed as input parameters for determining the sample rate for creating 3D strokes. A combination of an adaptive sampling rate and smoothing reduces stroke overlapping, improving overall aesthetics as shown in Figure 4. An adaptive sampling rate adjusts the hand position capture rate based on how fast the hand moves: the faster the hand movement, the higher the capture rate, up to a maximum of 60 Hz on a Google Pixel 4. The minimum capture rate is set to 5 Hz, the window size used for further smoothing. Note that the capture rate does not affect the main AR rendering FPS.

Smoothing can help improve stroke quality while reducing the effect of vision-based tracking noise [3]. While smoothing algorithms such as moving averages or Bézier curves can smooth out the resulting strokes, these smoothing algorithms prevent the user from sketching sharp corners, such as those generated by back-and-forth hand motions. We used a moving average with a window size of 5 to smooth every 5 raw position points at run-time. The result helps to improve not only the overall line quality but also sharp edges (Figure 4b).

3.2.2 Depth Indicators. Due to depth perception challenges from smartphone 3D free-hand interactions, users struggle when trying to align strokes in 3D space [37, 54]. Corrective techniques such as snapping have been adapted for aligning strokes to virtual objects [37]; however, these methods attach the current stroke to the endpoint of a single stroke but not along other positions of the existing stroke. In contrast, Portalware uses a depth indicating shader linked to the drawing index finger that appears when the user's finger intersects a stroke, highlighting all intersecting points along the stroke for visual feedback.

3.2.3 Free-hand Sketching. As previously mentioned, free-hand sketching allows the direct creation of AR strokes in 3D space with hand movements and gestures. Due to the smartphone's form factor, users need to hold the phone up with one hand and gesture (sketch) with their other hand. We implemented a set of onscreen user interface tools for single-hand use with the non-drawing hand to control the stroke's style, color, and width (Figure 3). This allows users to adjust the stroke's properties while continuing to perform hand gestures for sketching.

Free-hand interaction also allows users to interact with the spatial 3D UI that resides in the AR scene. Unlike onscreen 2D UI, 3D UI no longer stays at a fixed location on the screen, behaving like other AR objects in the scene. Prior small scale exploration with 3D UI note their increased immersion compared to onscreen UI [5]. We included a toggle button to let users to switch between 2D and 3D UI to develop their own 3D sketching workflow.

We tested both a pinch and a point gesture when designing different interaction schemes to minimize the "live-mic" issue. The "live-mic" issue occurs when it is difficult to determine the beginning or ending of an interaction. For gesture-based applications, the "live-mic" issue occurs when indicating the start or end of a gesture. When transitioning from a pinch to an idle gesture, the user creates unwanted strokes due to a non-instantaneous interaction state change (Figure 4c). Ultimately, a multimodal interaction in which the user sketches with one hand with a pointing gesture while using their other hand to tap-and-hold an on-screen button successfully triggers the state change instantly, eliminating the "live-mic" issue.

3.3 Dual-display Smartphone-Wearable Sketching

Our explorations of dual-display interactions are informed by single-display autobiographical design reports and the potential benefits of adding a second display. As more people carry two mobile displays (smartphone and smartwatch) in their daily lives [50], we hope to elicit genuine needs for a dual-display setup in mobile AR sketching.

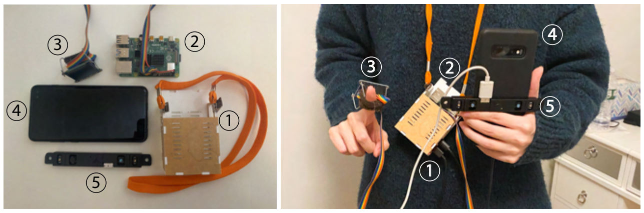


Figure 5: The Portalware hardware setup (left) consists of (1) a 3D-printed case, (2) a Raspberry Pi 4, (3) a SSD1351 OLED display, (4) a Galaxy S10+, and (5) a dual-fisheye depth camera for hand tracking. They can be assembled (right) to create a fully mobile experience.

While we describe the design lessons from using single-display sketching in Section 4.3.4, we want to explore if a dual-display sketching experience can help address some limitations from the single-display experience. For example, users must move their body to see occluded strokes on a smartphone, but may benefit from a second display on the finger, as the sketching hand does not have any form of feedback when it goes beyond the smartphone's view but is still within tracking range. In addition, the smartphone screen has very limited space for the UI interface, so we can explore whether the additional display is useful for accessing 3D UI elements.

3.3.1 Design Challenges. We started by designing the engineering pipeline and building a functional prototype (Section 3.3.3), iteratively testing and updating the system as we continued our ongoing interaction exploration. The authors encountered the following design challenges:

- The stroke rendered on the wearable display may not be legible given the small display dimensions.
- As the sketch becomes more complicated, existing strokes can overwhelm the wearable display, making it hard for the user to distinguish what is being drawn.
- The wearable does not have its own AR background by default.
- Since the wearable is mounted to the gesturing hand, hand movements can make the wearable display's content challenging to follow.

The following subsections explain how our implementation met these challenges experienced during the autobiographical design protocol.

3.3.2 Hardware Configuration. To maximize the hand tracking range for dual-display sketching, Portalware uses an ultra-wide (160° horizontal and vertical view range) dual-fisheye depth camera for hand tracking mounted on the back of the smartphone. This hand tracker runs on the same Orion 4.0 SDK that supports Leap Motion with equivalent tracking fidelity, but enables a wider tracking range. It detects hand motions and gestures beyond the field-of-view of the smartphone AR camera, allowing the user to sketch in on a larger canvas.

The wearable's electronic components (a 1.27-inch Adafruit SSD1351 OLED display, a Raspberry Pi 4, and a 2500 mAh battery) are enclosed by a wearable 3D printed case to ensure that the setup is fully mobile (Figure 5). We used a Raspberry Pi 4 since its Quad core CPU provides sufficient computation to handle data transmission with its on-board WiFi receiver; in addition, its built-in hardware SPI supports image data transmission to the SSD1351 OLED display, which has 128×96 pixel resolution in 32-bit color. Although a larger display may provide higher display quality, it adds additional weight on the fingertip and can also harm the hand's dexterity during mid-air sketching.

Since the wearable is an additional device not typically included in free-hand tracking models, we decided to examine how its mounting location affects hand tracking accuracy. During our pilot testing, we found that mounting the wearable on the back of the hand generally reduces tracking fidelity while mounting the wearable on the finger/fingertip did not seem to do so. Ultimately, we noticed that when the wearable is mounted on the back of the hand or wrist, it can be disorienting to view the display, as the user's gaze follows the wearable mounting location despite strokes extruding from the fingertip. This made it more difficult to mentally link the AR contents between the two displays. To minimize this distortion between stroke extrusion and the display's view, we decided to mount the wearable on the fingertip, expanding upon prior work on fingernail and finger-mounted displays [56, 61, 73].

3.3.3 System Pipeline. Extending the display content and interactions to the wearable requires communications between the smartphone and wearable in real-time. In order to support low-latency and fluid gestures, we created a sub-communication pipeline (Figure 6) to allow the wearable to receive display data with a communication latency of under 33 ms. The Raspberry Pi's on-board hardware SPI then decompresses and display the data stream at a 60 Hz refresh rate on the wearable. Since the primary role of the wearable is to extend the AR scene from the smartphone, this pipeline focuses on the communications from the smartphone to the wearable. In practice, the bottlenecks of this pipeline are *network*, *processing* and *display* speed.

For 128×96 resolution, the *network* and display bandwidth requirements are about 2.9 Mbits per second and 11.9 Mbits per second, respectively. Our network bandwidth requirements are difficult for the Pi's on-board Bluetooth transmitters to handle; therefore, we use a WebSocket protocol for wireless network transmission. In practice, we found that pixel data without compression can result in over a 2,000 ms delay when transmitting over WebSockets. Further encoding compression may be able to further reduce the size of the data stream and network load (the current packet rate per frame is currently 16.66 ms).

Once data is received over the network, the Pi *processes* data decompression and rescaling before displaying the data stream on the wearable. This requires GPU acceleration: we found that CPU-based loop structures in Python can only reach about 5 FPS after rescaling. We later modified the Pi's framebuffer to automatically scale up the entire display output, bypassing the need to upscale the data with loop structures, reaching a maximum of 60 FPS for the processing speed.

The final step of the communications is to *display* pixels on the wearable's OLED display. The Pi offers hardware SPI buses for high-speed communication (15 to 20 MHz) in theory. This speed is affected by the Pi's CPU load. Instead of reading off the pixels from memory after the previous step, we used the fbcp-ili9341 library to directly process pixel data from the GPU and communicate over hardware SPI [33].

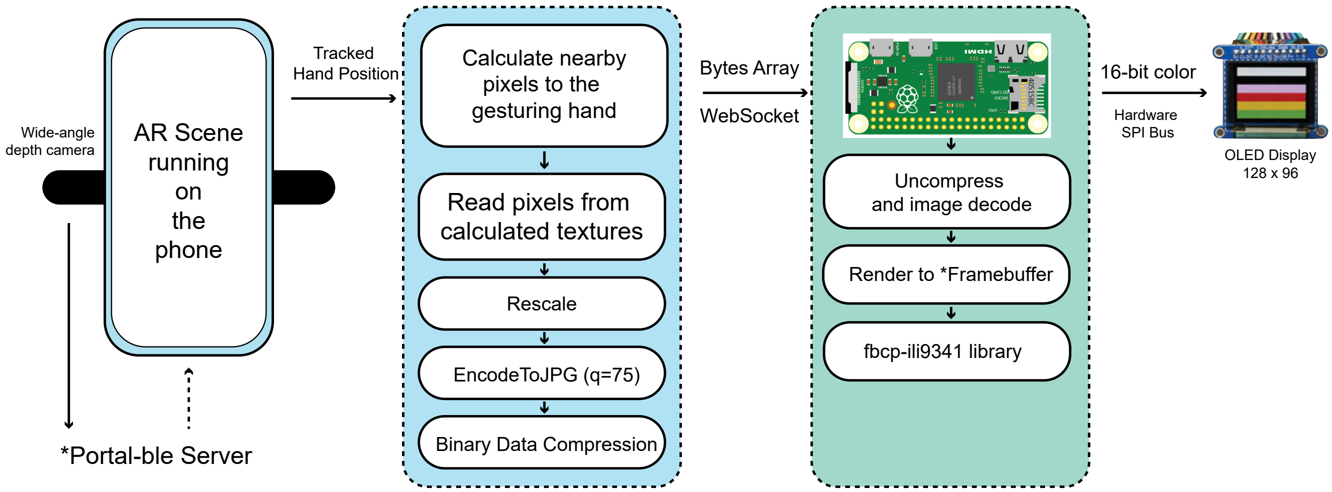


Figure 6: The system pipeline illustrates the synchronization of the smartphone's AR content with the wearable. *The Portal-ble server is a wearable computer that provides hand tracking computations while retaining the users' mobility.

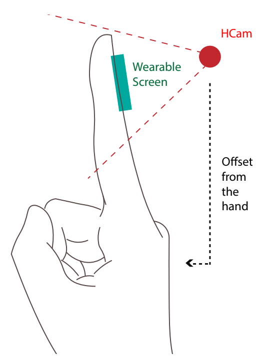
Overall, updating one frame of the AR data from the smartphone to wearable display takes about 32.7 ms between frames, or about 30 FPS on the wearable display.

3.3.4 Rendering on the Wearable Display. Since the wearable display lacks its own camera, we compute a simulated camera view (*HCam*) at the wearable's spatial location to render AR strokes on the wearable display. Due to the wearable's limited display size and resolution, we render the stroke sections close to the index fingertip. Initial testing revealed that the placement of *HCam* affects perception of the rendering results. For example, when virtually mounted to the back of the drawing index finger, *HCam* shows the endpoint of the stroke being drawn; when virtually mounted to the back of the palm, *HCam* shows the stroke segment. Note that the user's finger bends and rotates much more when making sketching gestures in comparison to the palm: mounting *HCam* to the finger results in frequent camera rotations. Therefore, we virtually mount *HCam* to the *palm* for stability, adding extra offsets on the Y axis to align *HCam* with the fingertip for better viewing consistency (Figure 7a). This method captures the stroke's shading, color, and texture according to the *HCam*'s view. The final results can be accessed via a GPU texture for efficiency.

To emphasize the stroke being currently drawn, a toggle switch for rendering on the wearable display makes existing strokes in the scene almost transparent by changing the opacity. In addition, post-processing options can enhance the stroke color's contrast, saturation and contour sharpness on the wearable display to increase perception via peripheral vision, as color perception decreases for peripheral vision [29].

Lacking a physical background environment, the wearable display uses the index finger's position and pointing orientation to determine the desired AR background region. While the wearable could simply duplicate the entire smartphone AR background, it does not account for the position or orientation of the index finger in the overall environment. Figure 7b shows how *HCam* follows the position and rotation of the index finger to "crop" a section of the raw AR background from the smartphone camera, correcting the perspective with the *HCam*'s projection.

Although *HCam* mounted to the palm is much more stable than when mounted on the finger, hand rotations during sketching can allow the *HCam*'s view to go beyond the smartphone's AR background texture (e.g. rotating the hand back and forth by 90°, as shown in Figure 8: Middle), making it impossible to find a region to render. In this case, the AR background on the wearable will be empty and fall back to a default black color. To avoid this exception, we use a rotation constraint remapping method applied to a Kalman filter



(a) The transformation of the palm is applied to the virtual camera (*HCam*) for better stability. An additional offset is used to align the *HCam* with the wearable display to achieve a consistent view during sketching.

(b) Due to the positioning of the wearable relative to the smartphone, each display shows a different orientation of the physical world. The wearable on the user's index finger provides the position and pointing orientation for determining the desired AR background regions.

Figure 7: A wearable display on the index finger enables a dual-display sketching experience in mobile AR.

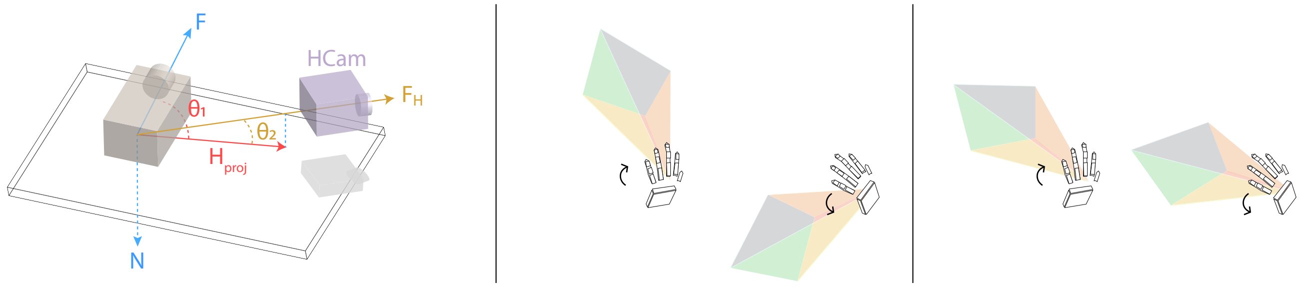


Figure 8: *Left*: The forward and right vectors form a plane, from which the forward direction is defined, and used to stabilize the view. *Middle*: *HCam* rotates excessively when the user rapidly moves their hand. *Right*: A remapping method and a Kalman filter helped to constrain this rotation.

to limit *HCam*'s rotation (Figure 8). First, we define a plane formed by the smartphone AR camera's *forward* and *right* vector. We then construct $\vec{H}_{proj}$, which is formed by projecting $HCam's$ forward vector onto this plane,

$$\vec{H}_{proj} = \vec{F}_H - (\vec{F}_H \cdot \vec{N})\vec{N} \tag{1}$$

$\vec{F}_H$ represents *HCam*'s forward vector, and $\vec{F}$ and $\vec{N}$ represent the AR camera's forward and normal vector of the plane (Figure 8). θ_1 denotes the angle between H_{proj} and F; θ_2 denotes the angle between F_H and H_{proj}. Then, we compute the two angles θ_1 and θ_2 that we use to constrain *HCam*,

$$\theta_1 = \arccos\left(\frac{\vec{H}_{proj} \cdot \vec{F}}{\left\|\vec{H}_{proj}\right\| \left\|\vec{F}\right\|}\right) \tag{2}$$

$$\theta_2 = \arccos\left(\frac{\vec{F}_H \cdot \vec{H}_{proj}}{\left\|\vec{F}_H\right\| \left\|\vec{H}_{proj}\right\|}\right) \tag{3}$$

Hence, we limit the rotation of θ_1 and θ_2,

$$\theta' = \theta \frac{2R}{\pi} \tag{4}$$

where R is the maximum raw rotation allowed for the constraint. We use θ_1' and θ_2' to compute *HCam*'s new orientation,

$$\vec{P} = \vec{F}\cos\theta_1' + \vec{C}\sin\theta_1' \Longrightarrow \vec{F}_{new} = \vec{P}\cos\theta_2' + \vec{N}\sin\theta_2' \tag{5}$$

$\vec{C}$ is calculated as the normalized cross product of $\vec{F}$ and $\vec{N}$. Note that $\vec{C}$ and $\vec{N}$ can each have two different directions, so we choose the directions which have a non-negative dot product with $\vec{F}_H$. Finally, we apply the $\vec{F}_{new}$ to *HCam* to set its new orientation. This enables the virtual camera to rotate much slower with stable rotations.

4 AUTOBIOGRAPHICAL DESIGN

To explore the design, nuanced interactions, and long-term usage challenges of smartphone free-hand AR sketching, we employed an autobiographical design protocol to iteratively develop, document, and test our system. Autobiographical design has been used in HCI to conduct research on the long-term relationships between users and systems [12]. This focus on auto-ethnography has become more prominent in situations that involve intimate devices such as wearable technology [9], and enables a deeper understanding of both small and large-scale effects on a system by placing the author in the roles of both designer and user [16, 46].

At its core, the autobiographical design protocol allows authors to intimately and directly experience a system from the user's perspective. Living with the system, authors engage with long-term AR sketching to rapidly gather design and engineering insights for updating the system, enabling first-hand documentation during the ongoing COVID-19 pandemic. Since interaction designs with smartphones or wearables are mostly underexplored for free-hand AR sketching, the autobiographical design protocol guides the authors to better experience and explore the underlying designs challenges with the author-user identity.

4.1 Author Background and Goals

Three authors of this paper participated in the autobiographical design protocol. We make each author's intentions explicit, to identify the tension between their role as designer and user in such autobiographical design research [16], and "describe their entering beliefs and biases" [14]. Each of the three authors has formal training in art and/or design, as well as computer science. Author A is currently focusing on computer science, but has multiple design degrees, including a BA for visual communication and an MFA for generative design. Author A believed that additional visual feedback should be useful for sketching but was unsure in what ways. Prioritizing precision and potential for everyday use, they primarily wanted to test the AR system's accuracy and see what they can sketch in a household setting. Author B has a painting background with a later focus on computer science. As they were familiar with the comfort of traditional mediums, they were more skeptical of the practicality of the smartphone-wearable approach and wanted to see if they could replicate the ease of experience of traditional drawing onto AR drawing. Author C has a background in both design and computer science. Author C was experienced in creating technical drawings requiring accurate measuring of object dimensions. Since AR overlays virtual objects directly over physical ones, they were interested in exploring how AR drawing can be benefited from using physical objects as references. Moreover, they were also curious about using the smartphone-wearable format to sketch. The authors' backgrounds allowed them to both think about the drawings from an artistic point of view, as well as consider system changes that could help them improve their 3D sketching.

We aim to identify key design lessons (i.e., what Neustaedter and Sengers [47] call "big effects") for free-hand AR sketching on smartphones in two formats: a smartphone AR format alone and a smartphone-wearable dual-screen format. Through these two formats, we explore interaction, perception, and design challenges for the authors, as well as insights to inform future designers thinking about AR sketching on mobile platforms. Although a smartphone-wearable format was found beneficial as haptic stimuli [52] or as an alternative input method [21], this format has yet to be explored as an auxiliary visual feedback device. Therefore, the authors examine whether such format can be useful in creating AR sketches, how this experience differs from a single-display format, and what design implications can arise from the auxiliary visual feedback device.

4.2 Methodology

Our autobiographical design process followed Neustaedter's five tenants [47]: *genuine needs*, *real systems*, *fast tinkering*, *record keeping and data collection*, and *long-term use*. These five tenants were used in prior work to help authors document and elicit design insights that are otherwise difficult to collect in a short time period. Our exploration began with genuine needs to understand the interaction implications using a working free-hand sketching system. Since *genuine needs* could comprise of an array of goals such as functionality or personal curiosity [16], the three authors approached this tenant from their own personal perspectives to understand interaction implications.

Authors documented their progress with screenshots, videos, and written notes to record their observations and organize insights. Screenshots and video recordings were only created when the author felt it was necessary and interesting, such as when they reached a personal milestone or discovered a new behavioral or perceptual challenge. Written notes about their overall experience were recorded every time after authors finished using the system. Based on qualitative analysis principles, each author reviewed their video recordings and written notes to extract key features indicating usability, perception, effort, and interaction challenges to form design lessons. Common themes that authors encountered were combined and summarized.

In this section, we first document three authors' stories developed from their recorded data, followed by design lessons they found important for the single-display format. Authors later explored a smartphone-wearable display format and similarly document elicited design lessons.

4.3 Single-Display Sketching Explorations

The authors started by building an AR sketching application on *Portal-ble* [54], an existing open-source system for free-hand mobile AR interactions, to identify challenges and limitations. Their personal experiences were then assembled into guidelines to inform design choices for Portalware.

4.3.1 Author A. As a first exercise, Author A scribbled different shapes, lines, emojis, and funny faces (Figure 9). They described

Figure 9: Virtual 3D sketches produced by Author A, showing illustrations with a multitude of fine-grained strokes representing household items.

the experience as enjoyable, like "drawing on an infinite canvas." Sometimes, when the sketching hand reached too far beyond the smartphone's view, Author A was unsure whether a stroke had been made, and could only check by physically moving and rotating the device. Author A then tried sketching objects of different sizes, spanning from tiny coins and flowers to mid-sized cups to larger objects like chairs. They noted that the smaller sketches were not very successful as the strokes tend to collapse together due to hand tracking errors. This problem was not observed for mid-size and larger items.

During their first few attempts, Author A noticed that sketching in 2D is much easier than in 3D even in 3D space: for example, spatially illustrating 2D patterns such as a star or rectangle is much easier than drawing a cube. It was easy to draw the front face of a cube, but not the other five faces. Lines that move away from the author (like performing a push action) are challenging to align with horizontal or vertical lines without additional assistance. This misalignment visually affects how the endpoints look as they do not fully overlap. Author A then tried the depth indicator and was able to connect the endpoints to draw the cube. More complex sketches with the depth indicator were also explored, including a vase and a 3D plant.

Author A noted that it was difficult to create long strokes that spanned across a room. They attempted to sketch a larger 6 ft $\times$ 6 ft cube across the room and were unable to align the edges across the room. Author A noted that when moving across the room while drawing, their gaze focused on their drawing hand to ensure that the lines were still extruding properly; however, upon examining their creation afterwards, they noticed that the cube's edges all had a zigzag shape. Upon more frequent usage, Author A noticed that sketching along the $x-y$ plane (vertical and horizontal direction) from the smartphone's perspective is much easier than in the z direction. They found that they could sketch more satisfactory long strokes by moving their body and sketching along the $x-y$ plane.

4.3.2 Author B. When initially testing the system, Author B reported growing strain in the arm holding the phone, rendering continuous drawing sessions difficult. To smooth this process, they gradually developed a steady workflow loop of sketching for 1 minute then resting for 5 seconds on repeat, and was able to create more complicated drawings through this method. To accommodate this loop, new load and save object functions were added to the system. These objects were anchored to a virtual cube, which could be grabbed and re-positioned by the user.

As they familiarized themselves with the system, Author B developed some personal habits for free-hand drawing: they maintained a steady "draw" pose, with the index finger held firmly upwards while maintaining the other fingers in a fist-like grip, to produce consistent lines. Different variations were additionally adapted for different types of drawings: single-planar (flat) drawings were easier and best created by keeping the hand holding the phone static while moving the drawing hand in all x, y, and z directions, creating an image only fully discernible from the perspective of the phone view. This method allowed Author B to create anamorphic drawings and optical illusions that only revealed their intended forms at specific angles in the virtual space. In these types of drawings, they found the function that hides previous strokes to be distracting and even confusing, since they could not see the overall context of their drawings. More difficult, time-consuming multi-planar drawings were created by first establishing a center point, then revolving the phone hand around that point. This method allowed the author to create fully-formed 3D drawings. In these cases, Author B found the ability to hide previous strokes to be more useful, especially in situations that required drawing behind virtual objects. These multi-planar drawings were further complemented by the introduction of the highlighting shader that indicates depth, which was particularly useful when drawing strokes attached to previous ones.

Throughout the drawing process, Author B also experienced difficulty with color swapping, noting the inconvenience of moving the drawing hand back to the 2D display each time to choose a color. The introduction of a 3D UI ameliorated this problem, as they could now directly manipulate hues in-place with less necessary motion.

4.3.3 Author C. To test the system, Author C began by drawing a series of simple 3D geometric objects and desk items, such as cubes and paper stationary, before progressing to larger and more complex objects, such as furniture. Due to its free-form nature, Author C noted how free-hand drawing worked best for drawing objects with organic and curvy forms, like plush toys and letters. Drawing larger objects like tables and chairs was sometimes physically tiring, as they had to physically move to draw strokes at specific positions; in addition, this larger range of motion resulted in more frequent AR tracking errors.

During testing, Author C encountered the "live mic" issue [72], finding that the movement of their fingers into and out of the pinching gesture made it difficult to predict stroke start and end positions. Furthermore, the pinching gesture was prone to tracking loss when facing away from the dual-fisheye depth camera. In contrast, they reported that the multimodal drawing mechanism was more effective, as the static pointing gesture made it intuitive to foresee the start and end positions of strokes. In addition, the pointing gesture was easily detected at more angles.

Besides drawing gestures, Author C had difficulties with compromised depth perception, which made it hard to position new strokes in relation to existing strokes. This often resulted in drawings that looked good in one angle but completely unrecognizable in another, and was mitigated with the depth indicator. Author C reported that it became much easier to draw new strokes at targeted positions because the indicator provided reliable visual cues to indicate where the index fingertip was relative to existing strokes. For example, they were able to more efficiently draw cubes that looked coherent when viewed from all angles.

Regarding the user interface, Author C reported that the 3D color picker was intuitive to learn and use because its interaction pattern was similar to drawing. In comparison to the more traditional on-screen 2D color picker, the 3D version provided a larger surface, allowing the author to more precisely adjust the hue by moving their gesturing hand.

After some drawing experiments, Author C developed their own workflow of creating 3D drawings. First, the author found real-life objects as references for dimensions and color, adjusting the stroke radius to the maximum size appropriate for the object. Then, they drew strokes to define the boundaries before filling in the boundaries with repeated parallel strokes. Author C also practiced drawing the same object several times to improve efficiency to reduce fatigue and minimize the impact of tracking errors.

4.3.4 Design Lessons for Free-hand Smartphone Drawing. Each of the three authors retrospectively reviewed their own recordings and written notes to extract key lessons in their autobiographical stories, engaging in the critical process of reflection [12]. They identified and emphasized different needs and challenges based on their personal experiences, summarized below.

- **The limited screen size of the mobile display restricts the freedom of free-hand interactions**. The dual-fisheye depth camera tracks a wider range than that depicted by the mobile display. This discrepancy between the size of traceable input and the size of visual output may be disorienting, coercing the user to follow the drawing hand with the hand holding the mobile device to see what they are drawing, restricting overall user movement.
- **Sketching in the plane perpendicular to the mobile display view while moving led to a loss of spatial awareness**. In particular, Author A reported difficulty creating straight lines since they could not follow the endpoint of the stroke as it was being drawn.
- **In mobile AR, creating flat drawings requires less time and physical movement in comparison to well-formed 3D drawings**. A possible explanation for this discrepancy is that flat drawings maintain the consistency of "2D input" to 2D output (the mobile display's view) due to their planar compositions. In contrast, creating 3D drawings in mobile AR involves drawing across all 3D dimensions while only being able to view a single perspective via the mobile display.
- **3D user interfaces provide greater ergonomic benefits and precision compared to 2D interfaces**. 2D UI elements are limited in placement and size by the mobile display, requiring the user to manipulate its functions on the small space they occupy. Both Authors B and C found that transitioning to a 3D UI expands this available space to the larger 3D canvas, removing the necessity to repeatedly move the drawing hand back to the smartphone screen.
- **When faced with drawing challenges, users created their own personalized solutions**. In particular, Authors B and C developed new respective workflows to combat the limitations they saw in the AR drawing framework, focusing on precise hand gestures to create accurate strokes and referencing real-life objects for dimension and color, respectively. A potential solution to possible drawing challenges in AR should also accommodate a large variety of possible workflows.

4.4 Dual-Display Sketching Explorations

The design lessons elicited from the single-display framework informed the development of the new dual-display Portalware system, described earlier in Section 4.3.4. At each stage of Portalware's technical development process, the three authors tested the system, drafting their own autobiographical stories from their personal perspectives and using these experiences to inspire future updates to the system. The entire iterative process, from its inception to the current version of Portalware, lasted a total of 14 weeks.

4.4.1 Author A. When sketching with the smartphone-wearable setup, Author A noted that the rendered content (AR objects and existing strokes) felt overwhelming for the small wearable display, sometimes leading to confusion about what was being shown. They found that the wearable display's content also moved too fast relative to the smartphone's AR background, which contributed to a loss of spatial reference in the AR environment. In the next few weeks, the wearable device went through a series of updates that included highlighting the actively drawn stroke while lowering opacity for all other existing strokes. These updates helped Author A better understand what is currently being drawn. With a less distracting wearable background, Author A could clearly see the stroke being drawn and an overall larger perceived sketching canvas. Author A likened this experience to the effect of persistence of vision [41], noting that the wearable display colors became more visible.

The wearable can change background color to indicate depth, but these colors appeared muted to Author A when viewed through the smartphone's display. Therefore, it was difficult for them to distinguish color changes easily when moving their drawing hand with the wearable out of and back into the smartphone's view. However, Author A did find the wearable useful when moving the hand parallel to the smartphone: since the display colors were more consistent through peripheral vision, they could immediately tell the hand's position in relation to the dual-fisheye depth camera's tracking range.

Later on, Author A tried to sketch next to physical objects in the same 6 ft × 6 ft room. The first few trials focused on making virtual augmentations, such as virtual bottles placed next to real bottles. Author A found it difficult to initially align their strokes to physical objects, mostly because of challenges distinguishing physical objects' depth on the smartphone display. Author A then developed a new stroke alignment strategy by leveraging the affordances of physical objects (e.g., touching a table surface or a lamp) and by looking directly at the wearable display on the finger when touching a physical object. This ensures that strokes always begin in alignment with the physical object before the user focuses back on the smartphone display to continue sketching. Ultimately, this strategy helped Author A align their strokes with physical objects.

4.4.2 Author B. With the introduction of a second wearable display attached to their right index finger, Author B found that the drawing time required for multi-planar drawings (Figure 10) decreased, especially when drawing occluded strokes. For drawing a simple skeleton of a sphere, the wearable roughly reduced drawing time from 95 seconds to 55 seconds. The author also commented that the unobstructed viewport for the drawing point provided by the wearable allowed them to directly connect strokes behind existing ones without physically re-positioning their body behind the virtual drawing to gauge depth. However, for single-planar drawings, they noted that the wearable display did not improve the drawing workflow and was at times distracting and superfluous. As an experiment, Author B also tried sketching without looking at the phone at all, relying solely on the wearable display. They reported that this was very difficult; due to the small size of the display and the inconsistency of the AR content movement with respect to the physical background, they could not see what they were drawing and pressed the *undo* button approximately four times more frequently.

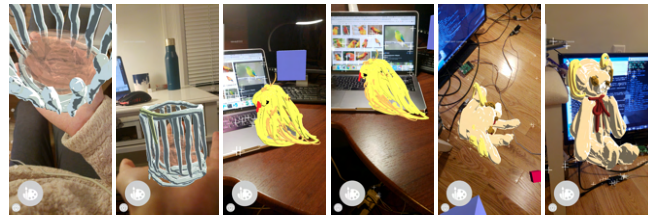


Figure 10: Samples from Author B's "multi-planar" drawings showing different angles of the same drawing.

Figure 11: Author B's drawings that attempt to complement the existing physical objects in the environment.

After adapting to the dual-display format, Author B developed a habit of "gaze-shifting." By purposefully switching gazes between the smartphone screen and wearable display, the latter of which consistently showed the drawing point even beyond the bounds of the smartphone screen, the author felt that the scope of their "perceptual canvas" was expanded. They noted that this notion could be further enriched with the addition of haptics, which can provide an additional dimension to interactions and alternative ways to delineate strokes and tracking ranges. They also reported that this expanded canvas effect was augmented by the addition of peripheral UI elements accessed through gestures with the wearable display, which made the smartphone screen less convoluted.

Author B felt that smartphone-wearable drawing was well complemented by the re-rendered physical background on the display, which was helpful in providing context for depth when tracing the forms of physical objects. Taking advantage of this re-rendered background shared across the two displays, Author B attempted to create drawings that enhanced the existing physical space. By sketching objects like a hanger, chair, and flower, they were able to create drawings that complement and enhance their environment (Figure 11). However, one limitation is that they can only interact with the exterior surfaces of each physical object, since the latter does not occupy the virtual space. While this limitation does not affect certain drawings, such as the sunflower, it does break immersion in the

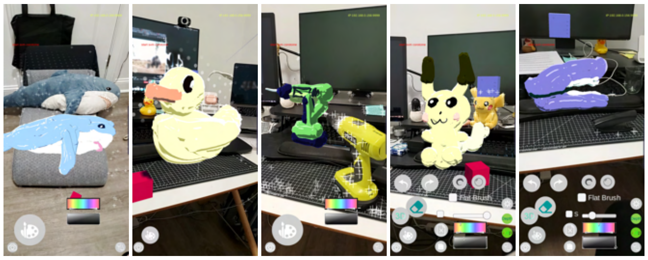


Figure 12: Author C's drawings that use real-life objects as references. The small white '+' symbols are tracking anchors that the author used while drawing.

sketch with a shirt on a hanger: the drawing visually appears to be *in front* of the physical object it is interacting with instead of being hooked on it, the latter of which requires greater imagination.

4.4.3 Author C. Author C found that the additional display improved their drawing workflow in two specific ways. The small portable display, because of its placement on the index fingertip, provided a different and more zoomed-in view of strokes occluded by other ones when viewed from the main smartphone screen. This made it easier for Author C to connect new strokes to existing strokes because they no longer had to move the phone or their body to see occluded strokes. This was especially helpful when drawing complex objects consisting of many parts like electronic tools and plush toys (Figure 12). Since fewer physical movements were involved, Author C also noted fewer tracking errors.

The most noticeable improvement that came with the introduction of the secondary portable display was the extension of the field-of-view. Author C stated that the single-display system setup was more suitable for drawing objects that appeared vertical because of the shape of the smartphone screen. Since the width of the display is narrower than its height, drawing objects wider than the smartphone display required constant movements of the smartphone in order to see and complete the strokes. In order to view the complete horizontal stroke, the author had to physically move back and hold the phone closer to their face. The secondary display, however, made it possible to see strokes drawn even when the hand moved beyond the smartphone screen's view. Author C was able to keep track of strokes without having to constantly move the phone to follow their hand. For example, when drawing a circle, Author C found it more efficient for circles to be as wide as the smartphone screen's width with the single-display setup, but could now more easily draw circles with the phone's height as the radius because of the extended field-of-view. They reported that they were able to draw a wider range of objects than before. One caveat is that it was hard to determine how far away from the smartphone they could move their hand from without encountering hand tracking issues.

When the fingertip was close to a stroke, most of the wearable display would convert to the color of that stroke. This could be helpful for the erasing function, as it indicated that the fingertip was touching the stroke. However, when Author C touched a black or dark-colored stroke, the small display would turn completely black, sometimes confusing the author.

4.4.4 Design Lessons for Dual-Display Sketching. In the second stage of autobiographical design, the authors focused on the dual-display sketching experience. Their stories revealed several implications integral to the smartphone-wearable framework:

- **Wearable use cases are specific to the type of drawing**. The wearable display is only helpful during instances of occlusion for multi-planar drawings, and offers little benefit for single-planar drawings.
- **A wearable small enough to be mobile is inadequate to serve as a primary display**. It is difficult to draw objects when relying solely on the wearable display due to its (1) small size and (2) rapidly moving content, which may disorient users. Therefore, we recommend that small wearable devices, such as those mounted to the finger or fingernail, focus on sensing input methods instead of visualizing content on a display.
- **Adding depth indicators on a wearable display is helpful for indicating tracking range**. Since the wearable display is mounted to the drawing finger, which the user's gaze follows as they draw, they immediately notice when the finger falls outside the tracking range. These depth indicators provide additional visual feedback to the display's rendered content, providing an additional affordance for interacting with 3D strokes.
- **Mounting a wearable display to the fingertip helps reduce the cognitive gap between input and output locations**. Initially, the authors experienced disorientation due to the system's separation of the location of visual input (the drawing finger) and location of visual feedback (the drawing), two factors that usually occupy the same space in traditional drawing settings. Reuniting the two again by allowing the user to shift their gaze between the phone and finger-mounted wearable display (as opposed to a smartwatch) can lead to more enjoyable drawing experiences.
- **A wearable display can expand the "perceptual canvas."** While this observation is not formally verified, the authors reported that the wearable display ameliorated the lack of spatial awareness by increasing the perceived dimensions of the virtual canvas. Specifically, Author A noted that gaze shifting allowed them to develop a persistent spatial memory of the canvas, allowing them to better plan future strokes to fill up the drawing space. This increased spatial awareness can help make stroke-based interactions beyond the smartphone's view more precise along the periphery.

4.5 Limitations and Future Work

Following an autobiographical design protocol allowed us to make rapid iterative improvements to Portalware to tailor it towards our own artistic and design goals. However, although perhaps suitable for other users who have similar levels of experience in engineering and design as the authors, the resultant lessons from this approach lacked observations from users of both higher and lower skill levels, and is thus not generalizable [47]. For example, while Author B and Author C emphasized how they were able to develop a natural workflow with Portalware, a user who is less accustomed to free-hand drawing may desire more guidance from the system itself to inform them on how to arrive at such a workflow. Because the authors became very familiar with Portalware after long months of usage, we were unable to elicit insights into Portalware's learnability. Another limitation of adopting the autobiographical design paradigm to Portalware is the inability to record reliable and quantifiable metrics. The authors observed in their own cases that drawings were faster to complete and spatial awareness was improved, but proper human subject experiments could be conducted in the future to attribute significance to the degree of these improvements. Future work may supplement the autobiographical design with mixed methods involving such usability testing with a broader population, and evaluation of unfamiliar users learning the interaction techniques.

5 CONCLUSION

The autobiographical design protocol has guided our exploration of free-hand sketching from a single-display format to a dual-display one with the addition of a wearable to supplement the smartphone's view. This transformation over several months of use, resulting in the Portalware system for free-hand AR drawing, has addressed some of the learning challenges with 3D sketching [71]. We tell short narratives of three authors, each with different artistic backgrounds and motivations, as they experiment with wearable placements to elicit usability lessons. We learn how this wearable can help indicate depth and stroke alignment, as users adjust to the idea of having a virtual canvas "in the air" rather than on a screen. While Portalware is not likely to be the final form of dual-display AR interfaces, these stories illustrate the limitations and potential directions beyond usability problems, and provide implications more generally for hand-held augmented reality devices. The resulting experiences and sketches provide evidence that there is potential for artists to produce work in the context of the physical environment, with devices that can be taken on the go.

ACKNOWLEDGMENTS

We thank Laura Wilson, Shaun Wallace, Fumeng Yang, and Hyeongjin Kim for reviewing the draft. We also thank Vincent LeQuang, Dobuki Studio on bandcamp, for his background music for our teaser video and Binglu Du for videotaping. This work is supported by the National Science Foundation under Grant No. IIS-1552663.

REFERENCES

[1] Hideyuki Ando, Takeshi Miki, Masahiko Inami, and Taro Maeda. 2002. SmartFinger: nail-mounted tactile display. In *ACM SIGGRAPH 2002 conference abstracts and applications*. 78–78. https://doi.org/10.1145/1242073.1242113

[2] Hideyuki Ando, Junji Watanabe, Masahiko Inami, Maki Sugimito, and Taro Maeda. 2007. A fingernail-mounted tactile display for augmented reality systems. *Electronics and Communications in Japan (Part II: Electronics)* 90, 4 (2007), 56–65. https://doi.org/10.1002/ecjb.20355

[3] Rahul Arora, Rubaiat Habib Kazi, Tovi Grossman, George Fitzmaurice, and Karan Singh. 2018. Symbiosissketch: Combining 2d & 3d sketching for designing detailed 3d objects in situ. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–15. https://doi.org/10.1145/3173574.3173759

[4] Rahul Arora, Rubaiat Habib Kazi, Fraser Anderson, Tovi Grossman, Karan Singh, and George W Fitzmaurice. 2017. Experimental Evaluation of Sketching on Surfaces in VR. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Vol. 17. 5643–5654. https://doi.org/10.1145/3025453.3025474

[5] Andrew Aultman, Spencer Dowie, and Nelly Ann Hamid. 2018. Design Heuristics for Mobile Augmented Reality Game User Interfaces. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–5. https://doi.org/10.1145/3170427.3188580

[6] Mihai Bâce, Gábor Sörös, Sander Staal, and Giorgio Corbellini. 2017. HandshakAR: wearable augmented reality system for effortless information sharing. In *Proceedings of the 8th Augmented Human International Conference*. 1–5. https://doi.org/10.1145/3041164.3041203

[7] M Barrera Machuca, P Asente, W Stuerzlinger, J Lu, and B Kim. 2018. Multiplanes: Assisted freehand vr sketching. In *Proceedings of the Symposium on Spatial User Interaction, SUI*, Vol. 18. 13–14. https://doi.org/10.1145/3267782.3267786

[8] Mark Billinghurst, Tham Piumsomboon, and Huidong Bai. 2014. Hands in space: Gesture interaction with augmented-reality interfaces. *IEEE computer graphics and applications* 34, 1 (2014), 77–80. https://doi.org/10.1109/MCG.2014.8

[9] Marta E Cecchinato, Anna L Cox, and Jon Bird. 2017. Always on (line)? User experience of smartwatches and their role within multi-device ecologies. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 3557–3568. https://doi.org/10.1145/3025453.3025538

[10] Yun Suk Chang, Benjamin Nuernberger, Bo Luan, Tobias Höllerer, and John O'Donovan. 2017. Gesture-based augmented reality annotation. In *2017 IEEE Virtual Reality (VR)*. IEEE, 469–470. https://doi.org/10.1109/VR.2017.7892383

[11] Xiang 'Anthony' Chen, Tovi Grossman, Daniel J Wigdor, and George Fitzmaurice. 2014. Duet: exploring joint interactions on a smart phone and a smart watch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 159–168. https://doi.org/10.1145/2556288.2556955

[12] Wei-Chi Chien and Marc Hassenzahl. 2020. Technology-mediated relationship maintenance in romantic long-distance relationships: An autoethnographical research through design. *Human–Computer Interaction* 35, 3 (2020), 240–287. https://doi.org/10.1080/07370024.2017.1401927

[13] Klen Čopič Pucihar, Paul Coulton, and Jason Alexander. 2013. Evaluating dual-view perceptual issues in handheld augmented reality: device vs. user perspective rendering. In *Proceedings of the 15th ACM on International conference on multimodal interaction*. 381–388. https://doi.org/10.1145/2522848.2522885

[14] John W Creswell and Dana L Miller. 2000. Determining validity in qualitative inquiry. *Theory into practice* 39, 3 (2000), 124–130. https://doi.org/10.1207/s15430421tip3903_2

[15] Shujie Deng, Nan Jiang, Jian Chang, Shihui Guo, and Jian J Zhang. 2017. Understanding the impact of multimodal interaction using gaze informed mid-air gesture control in 3D virtual objects manipulation. *International Journal of Human-Computer Studies* 105 (2017), 68–80. https://doi.org/10.1016/j.ijhcs.2017.04.002

[16] Audrey Desjardins and Aubree Ball. 2018. Revealing tensions in autobiographical design in HCI. In *proceedings of the 2018 designing interactive systems conference*. 753–764. https://doi.org/10.1145/3196709.3196781

[17] Arindam Dey, Graeme Jarvis, Christian Sandor, and Gerhard Reitmayr. 2012. Tablet versus phone: Depth perception in handheld augmented reality. In *2012 IEEE international symposium on mixed and augmented reality (ISMAR)*. IEEE, 187–196. https://doi.org/10.1109/ISMAR.2012.6402556

[18] Tobias Drey, Jan Gugenheimer, Julian Karlbauer, Maximilian Milo, and Enrico Rukzio. 2020. VRSketchIn: Exploring the Design Space of Pen and Tablet Interaction for 3D Sketching in Virtual Reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–14. https://doi.org/10.1145/3313831.3376628

[19] Ruofei Du, Eric Turner, Maksym Dzitsiuk, Luca Prasso, Ivo Duarte, Jason Dourgarian, Joao Afonso, Jose Pascoal, Josh Gladstone, Nuno Cruces, et al. 2020. DepthLab: Real-Time 3D Interaction With Depth Maps for Mobile Augmented Reality. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. 829–843. https://doi.org/10.1145/3379337.3415881

[20] 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*. 241–252. https://doi.org/10.1145/3196709.3196737

[21] Anton Fedosov, Evangelos Niforatos, Ivan Elhart, Teseo Schneider, Dmitry Anisimov, and Marc Langheinrich. 2016. Design and evaluation of a wearable AR system for sharing personalized content on ski resort maps. In *Proceedings of the 15th International Conference on Mobile and Ubiquitous Multimedia*. ACM, 141–152. https://doi.org/10.1145/3012709.3012721

[22] John Galeotti, Samantha Horvath, Roberta Klatzky, Brock Nichol, Mel Siegel, and George Stetten. 2008. FingerSight™: fingertip control and haptic sensing of the visual environment. In *ACM SIGGRAPH 2008 new tech demos*. 1–1. https://doi.org/10.1145/1401615.1401631

[23] Danilo Gasques, Janet G Johnson, Tommy Sharkey, and Nadir Weibel. 2019. What you sketch is what you get: Quick and easy augmented reality prototyping with pintar. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–6. https://doi.org/10.1145/3290607.3312847

[24] Daniele Giunchi, Stuart James, Donald Degraen, and Anthony Steed. 2019. Mixing realities for sketch retrieval in Virtual Reality. In *The 17th International Conference on Virtual-Reality Continuum and its Applications in Industry*. 1–2. https://doi.org/10.1145/3359997.3365751

[25] Leo Gombač, Klen Čopič Pucihar, Matjaž Kljun, Paul Coulton, and Jan Grbac. 2016. 3D Virtual tracing and depth perception problem on mobile AR. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. 1849–1856. https://doi.org/10.1145/2851581.2892412

[26] Google. 2020. TiltBrush. https://www.tiltbrush.com/

[27] Uwe Gruenefeld, Tim Claudius Stratmann, Wilko Heuten, and Susanne Boll. 2017. PeriMR: a prototyping tool for head-mounted peripheral light displays in mixed reality. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*. ACM, 51. https://doi.org/10.1145/3098279.3125439

[28] Jeremy Gummeson, Bodhi Priyantha, and Jie Liu. 2014. An energy harvesting wearable ring platform for gestureinput on surfaces. In *Proceedings of the 12th*

annual international conference on Mobile systems, applications, and services. 162–175. https://doi.org/10.1145/2594368.2594389
[29] Thorsten Hansen, Lars Pracejus, and Karl R Gegenfurtner. 2009. Color perception in the intermediate periphery of the visual field. *Journal of Vision* 9, 4 (2009), 1–12. https://doi.org/10.1167/9.4.26
[30] Wolfgang Hürst, Ronald Poppe, and Jerry Van Angeren. 2015. Drawing outside the lines: Tracking-based gesture interaction in mobile augmented entertainment. In *2015 7th International Conference on Intelligent Technologies for Interactive Entertainment (INTETAIN)*. IEEE, 79–87. https://doi.org/10.4108/icst.intetain.2015.259589
[31] Wolfgang Hürst and Casper Van Wezel. 2013. Gesture-based interaction via finger tracking for mobile augmented reality. *Multimedia Tools and Applications* 62, 1 (2013), 233–258. https://doi.org/10.1007/s11042-011-0983-y
[32] Hans-Christian Jetter, Roman Rädle, Tiare Feuchtner, Christoph Anthes, Judith Friedl, and Clemens Nylandsted Klokmose. 2020. " In VR, everything is possible!": Sketching and Simulating Spatially-Aware Interactive Spaces in Virtual Reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–16. https://doi.org/10.1145/3313831.3376652
[33] Juj. 2019. juj/fbcp-ili9341. https://github.com/juj/fbcp-ili9341
[34] Hsin-Liu Kao, Artem Dementyev, Joseph A Paradiso, and Chris Schmandt. 2015. NailO: fingernails as an input surface. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 3015–3018. https://doi.org/10.1145/2702123.2702572
[35] Junhyeok Kim, William Delamare, and Pourang Irani. 2018. Thumbtext: Text entry for wearable devices using a miniature ring. In *Graphics Interface*. 18–25. https://doi.org/10.20380/GI2018.04
[36] Ernst Kruijff, J Edward Swan, and Steven Feiner. 2010. Perceptual issues in augmented reality revisited. In *2010 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 3–12. https://doi.org/10.1109/ISMAR.2010.5643530
[37] Kin Chung Kwan and Hongbo Fu. 2019. Mobi3dsketch: 3D sketching in mobile AR. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–11. https://doi.org/10.1145/3290605.3300406
[38] Jung-hoon Kwon, Han-wool Choi, Jeong-in Lee, and Young-Ho Chai. 2005. Free-Hand stroke based NURBS surface for sketching and deforming 3d contents. In *Pacific-Rim Conference on Multimedia*. Springer, 315–326. https://doi.org/10.1007/11581772_28
[39] Germán Leiva, Cuong Nguyen, Rubaiat Habib Kazi, and Paul Asente. 2020. Pronto: Rapid Augmented Reality Video Prototyping Using Sketches and Enaction. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13. https://doi.org/10.1145/3313831.3376160
[40] Gravity Sketch Limited. 2020. GravitySketch. https://www.gravitysketch.com/
[41] Xin Liu, Katia Vega, Jing Qian, Joseph Paradiso, and Pattie Maes. 2016. Fluxa: Body movements as a social display. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, 155–157. https://doi.org/10.1145/2984751.2985741
[42] Fabrice Matulic, Riku Arakawa, Brian Vogel, and Daniel Vogel. 2020. PenSight: Enhanced Interaction with a Pen-Top Camera. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–14. https://doi.org/10.1145/3313831.3376147
[43] Daniel Mendes, Daniel Medeiros, Maurício 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)*. IEEE, 154–157. https://doi.org/10.1109/3DUI.2017.7893332
[44] SangHun Nam and JoungHuem Kwon. 2020. Virtual Reality Interface for Visual Art with a Five-Fingered Haptic Controller. In *Proceedings of the 2020 3rd International Conference on Image and Graphics Processing*. 101–105. https://doi.org/10.1145/3383812.3383840
[45] Suranga Nanayakkara, Roy Shilkrot, Kian Peen Yeo, and Pattie Maes. 2013. EyeRing: a finger-worn input device for seamless interactions with our surroundings. In *Proceedings of the 4th Augmented Human International Conference*. 13–20. https://doi.org/10.1145/2459236.2459240
[46] Carman Neustaedter, Tejinder K Judge, and Phoebe Sengers. 2014. Autobiographical design in the home. *Studying and Designing Technology for Domestic Life: Lessons from Home* 135 (2014), 135–158.
[47] Carman Neustaedter and Phoebe Sengers. 2012. Autobiographical design in HCI research: designing and learning through use-it-yourself. In *Proceedings of the Designing Interactive Systems Conference*. 514–523. https://doi.org/10.1145/2317956.2318034
[48] Shahriar Nirjon, Jeremy Gummeson, Dan Gelb, and Kyu-Han Kim. 2015. Typingring: A wearable ring platform for text input. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*. 227–239. https://doi.org/10.1145/2742647.2742665
[49] Masataka Niwa, Tomoko Nozaki, Taro Maeda, and Hideyuki Ando. 2010. Fingernail-mounted display of attraction force and texture. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*. Springer, 3–8. https://doi.org/10.1007/978-3-642-14075-4_1
[50] Obi Ogbanufe and Natalie Gerhart. 2020. The mediating influence of smartwatch identity on deep use and innovative individual performance. *Information Systems Journal* 30, 6 (2020), 977–1009. https://doi.org/10.1111/isj.12288
[51] Ju Young Oh, Ji-Hyung Park, and Jung-Min Park. 2020. FingerTouch: Touch Interaction Using a Fingernail-Mounted Sensor on a Head-Mounted Display for Augmented Reality. *IEEE Access* 8 (2020), 101192–101208. https://doi.org/10.1109/access.2020.2997972
[52] Gianluca Paolocci, Tommaso Lisini Baldi, Davide Barcelli, and Domenico Prattichizzo. 2020. Combining Wristband Display and Wearable Haptics for Augmented Reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE, 633–634. https://doi.org/10.1109/VRW50115.2020.00167
[53] Alex Peer and Kevin Ponto. 2019. Mitigating Incorrect Perception of Distance in Virtual Reality through Personalized Rendering Manipulation. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 244–250. https://doi.org/10.1109/VR.2019.8797911
[54] Jing Qian, Jiaju Ma, Xiangyu Li, Benjamin Attal, Haoming Lai, James Tompkin, John F Hughes, and Jeff Huang. 2019. Portal-ble: Intuitive Free-hand Manipulation in Unbounded Smartphone-based Augmented Reality. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. ACM, 133–145. https://doi.org/10.1145/3332165.3347904
[55] Farzad Pour Rahimian and Rahinah Ibrahim. 2011. Impacts of VR 3D sketching on novice designers' spatial cognition in collaborative conceptual architectural design. *Design Studies* 32, 3 (2011), 255–291. https://doi.org/10.1016/j.destud.2010.10.003
[56] Thijs Roumen, Simon T Perrault, and Shengdong Zhao. 2015. Notiring: A comparative study of notification channels for wearable interactive rings. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 2497–2500. https://doi.org/10.1145/2702123.2702350
[57] Danilo Alves Santos, David Szturm, Lays Xavier de Castro, Juliana Santos de Souza Hannum, and Talles Marcelo G de A Barbosa. 2017. Wearable device for literacy activities with people with down syndrome. In *2017 IEEE MIT Undergraduate Research Technology Conference (URTC)*. IEEE, 1–4. https://doi.org/10.1109/URTC.2017.8284204
[58] Jie Song, Gábor Sörös, Fabrizio Pece, Sean Fanello, Shahram Izadi, Cem Keskin, and Otmar Hilliges. 2014. In-air Gestures Around Unmodified Mobile Devices. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. 319–329. https://doi.org/10.1145/2642918.2647373
[59] Darko Stanimirovic and Daniel Kurz. 2014. Smartwatch-aided handheld augmented reality. In *2014 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 373–374. https://doi.org/10.1109/ISMAR.2014.6948495
[60] Lee Stearns, Victor DeSouza, Jessica Yin, Leah Findlater, and Jon E Froehlich. 2017. Augmented reality magnification for low vision users with the microsoft hololens and a finger-worn camera. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility*. 361–362. https://doi.org/10.1145/3132525.3134812
[61] Chao-Huai Su, Liwei Chan, Chien-Ting Weng, Rong-Hao Liang, Kai-Yin Cheng, and Bing-Yu Chen. 2013. NailDisplay: bringing an always available visual display to fingertips. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1461–1464. https://doi.org/10.1145/2470654.2466193
[62] Ryo Suzuki, Rubaiat Habib Kazi, Li-Yi Wei, Stephen DiVerdi, Wilmot Li, and Daniel Leithinger. 2020. RealitySketch: Embedding Responsive Graphics and Visualizations in AR through Dynamic Sketching. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. 166–181. https://doi.org/10.1145/3379350.3416155
[63] Paul Taele and Tracy Hammond. 2014. Developing sketch recognition and interaction techniques for intelligent surfaceless sketching user interfaces. In *Proceedings of the companion publication of the 19th international conference on Intelligent User Interfaces*. 53–56. https://doi.org/10.1145/2559184.2559185
[64] Julien Valentin, Adarsh Kowdle, Jonathan T Barron, Neal Wadhwa, Max Dzitsiuk, Michael Schoenberg, Vivek Verma, Ambrus Csaszar, Eric Turner, Ivan Dryanovski, et al. 2018. Depth from motion for smartphone AR. *ACM Transactions on Graphics (ToG)* 37, 6 (2018), 1–19. https://doi.org/10.1145/3272127.3275041
[65] Katia Vega and Hugo Fuks. 2014. Beauty tech nails: interactive technology at your fingertips. In *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction*. 61–64. https://doi.org/10.1145/2540930.2540961
[66] Sharad Vikram, Lei Li, and Stuart Russell. 2013. Writing and sketching in the air, recognizing and controlling on the fly. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*. 1179–1184. https://doi.org/10.1145/2468356.2468567
[67] Philipp Wacker, Oliver Nowak, Simon Voelker, and Jan Borchers. 2019. ARPen: Mid-Air Object Manipulation Techniques for a Bimanual AR System with Pen & Smartphone. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. 1–12. https://doi.org/10.1145/3290605.3300849
[68] 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 *Proceedings of the Symposium on Spatial User Interaction*. 25–35. https://doi.org/10.1145/3267782.3267788
[69] Philipp Wacker, Adrian Wagner, Simon Voelker, and Jan Borchers. 2020. Heatmaps, Shadows, Bubbles, Rays: Comparing Mid-Air Pen Position Visualizations in Handheld AR. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1–11. https://doi.org/10.1145/3313831.3376848

[70] Martin Weigel and Jürgen Steimle. 2013. Fingernail displays: Handy displays at your fingertips. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*. 937–942.

[71] 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*. 135–142. https://doi.org/10.2312/SBM/SBM10/135-142

[72] Daniel Wigdor and Dennis Wixon. 2011. *Brave NUI world: designing natural user interfaces for touch and gesture*. Morgan Kaufmann, San Francisco, CA. https://doi.org/10.1016/C2009-0-64091-5

[73] Raphael Wimmer and Florian Echtler. 2013. Exploring the benefits of fingernail displays. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*. 937–942. https://doi.org/10.1145/2468356.2468524

[74] Robert Xiao and Hrvoje Benko. 2016. Augmenting the field-of-view of head-mounted displays with sparse peripheral displays. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1221–1232. https://doi.org/10.1145/2858036.2858212

[75] Chun Yu, Xiaoying Wei, Shubh Vachher, Yue Qin, Chen Liang, Yueting Weng, Yizheng Gu, and Yuanchun Shi. 2019. Handsee: enabling full hand interaction on smartphone with front camera-based stereo vision. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13. https://doi.org/10.1145/3290605.3300935

[76] Boning Zhang, Yiqiang Chen, Yueliang Qian, and Xiangdong Wang. 2011. A ring-shaped interactive device for large remote display and mobile device control. In *Proceedings of the 13th international conference on Ubiquitous computing*. 473–474. https://doi.org/10.1145/2030112.2030177