

Pronto: Rapid Augmented Reality Video Prototyping Using Sketches and Enaction

Germán Leiva¹Cuong Nguyen²Rubaiat Habib Kazi²Paul Asente²

¹Digital Design and Information Studies, Aarhus University
leiva@cavi.au.dk

²Adobe Research
{cunguyen,rhabib,asente}@adobe.com

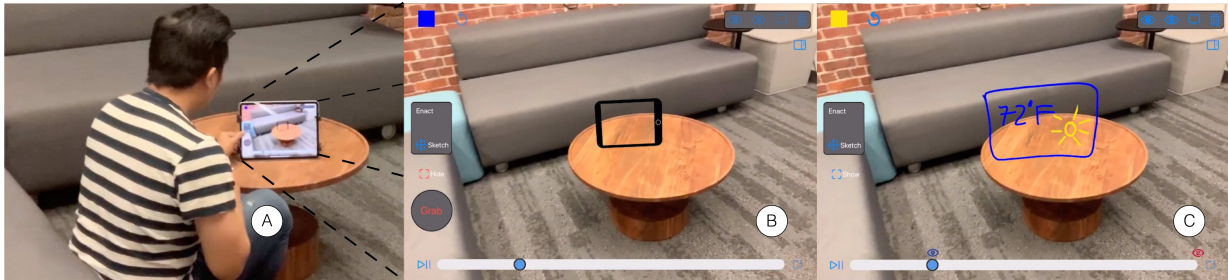


Figure 1. PRONTO's video prototyping workflow example. A designer wants to prototype an AR weather interface on a coffee table: First, the designer records a video scenario imagining the interface. (a) The designer physically places a tablet device running PRONTO on top of the coffee table, demonstrating the desired position of the interface. (b) The initially recorded video also captured AR data, which allows the designer to immediately see, within the recorded video, the tablet's location from the initial recording perspective. (c) Finally, the designer sketches at the defined position to quickly convey the weather interface. The resulting prototype is a video combined with graphics, all created with a single device and a rapid workflow.

ABSTRACT

Designers have limited tools to prototype AR experiences rapidly. Can lightweight, immediate tools let designers prototype dynamic AR interactions while capturing the nuances of a 3D experience? We interviewed three AR experts and identified several recurring issues in AR design: creating and positioning 3D assets, handling the changing user position, and orchestrating multiple animations. We introduce PROJECT PRONTO, a tablet-based video prototyping system that combines 2D video with 3D manipulation. PRONTO supports four intertwined activities: *capturing* 3D spatial information alongside a video scenario, *positioning* and *sketching* 2D drawings in a 3D world, and *enacting* animations with physical interactions. An observational study with professional designers shows that participants can use PRONTO to prototype diverse AR experiences. All participants performed two tasks: replicating a sample non-trivial AR experience and prototyping their open-ended designs. All participants completed the replication task and found PRONTO easy to use. Most participants found that PRONTO encourages more exploration of designs than their current practices.

Author Keywords

AR; Sketching; Video Prototyping; Design by Enaction

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

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

© 2020 Association for Computing Machinery.

ACM ISBN 978-1-4503-6708-0/20/04 ...\$15.00.

<http://dx.doi.org/10.1145/3313831.3376160>

CCS Concepts

•Human-centered computing → Mixed / augmented reality; Interface design prototyping; Systems and tools for interaction design; Empirical studies in HCI;

INTRODUCTION

Augmented Reality (AR) is becoming widely available [46], with new devices and services constantly appearing [9]. This emerging ecosystem enables richer user experiences [13], encouraging designers to explore varied AR designs before implementing them. Designers traditionally rely on rapid prototyping tools to explore alternatives in early-stage design for other tasks [5]. However, few of these tools provide features needed to support AR experiences [45], requiring designers to use alternative approaches.

Many popular AR authoring tools require 3D programming knowledge. Game engine environments like Unity and Unreal, and Software Development Kits (SDKs) like ARKit or ARCore, provide powerful tools to manipulate 3D assets and create an interactive AR prototype. However, the need for programming knowledge hinders exploration and excludes non-technical designers. Even for technical designers, programming slows down prototyping by requiring early attention to low-level details, such as visual fidelity, world-tracking, and gesture recognition. Creating these runnable interactive prototypes typically requires repetitive and time-consuming code-build-test cycles, which discourages exploration and can lead to design fixation [22].

Other authoring tools do not require coding, e.g., Spark Studio¹ and Lens Studio². However, they focus on a limited set of

¹<https://sparkar.facebook.com/ar-studio>

²<https://lensstudio.snapchat.com>

AR interactions, such as face-tracking, and generally require both a desktop for authoring and an AR device for testing. Other code-free tools like Torch³ combine the authoring and target platforms, but they provide only a limited set of animations and interaction triggers. Default interaction triggers, such as the user’s proximity or the device’s orientation, are only available as static menu options in the tool’s UI and cannot be used directly to create animations. This forced indirection between authoring and interaction design can hinder the ideation process of novel physical AR experiences.

Instead of building interactive AR prototypes, many designers explore and communicate new design ideas with *video prototypes*— non-interactive illustrative videos that include overlaid motion graphics. They start by recording a video of a scenario, generally in the environment where the real AR experience takes place [52]. Later, they use video editing tools like After Effects or Final Cut Pro to add motion graphics to the video [35]. This approach is more flexible than using programming-based tools, even letting designers illustrate experiences that they cannot create with current technology. However, it can be laborious, requiring techniques like rotoscoping and keyframe animation. The prototyping process is also indirect, requiring designers to create 3D experiences through a 2D interface on a desktop device.

Most commercial AR tools lack features that we consider important for creating a wide variety of prototypes. Some, like ActionGram⁴ and HoloStudio⁵, focus on passive and pre-made assets, either static or with fixed animations. This makes it difficult to prototype dynamic and reactive AR experiences on the fly. A goal of PRONTO is rapid creation and animation of assets on-the-spot in order to illustrate custom user interactions with the content. Other tools, such as Chalk⁶, support the rapid creation of assets through sketching, but these assets are static and designed to be annotations for remote collaboration rather than AR prototypes.

How can a single prototyping system provide the 3D support of programming, the flexibility of video prototyping, and the speed of specialized tools? To inform the design of such a tool, we conducted three interviews with expert AR designers to understand current issues in AR design. Based on these interviews and the previously described issues, we distilled four design goals for a new rapid prototyping tool:

1. create low-fidelity assets through sketching
2. position assets in space through direct manipulation
3. capture user position changes through video
4. create complex animations through direct manipulation

We designed and implemented PRONTO: a novel AR video prototyping system that combines 2D video with 3D manipulation.

PRONTO supports four intertwined activities to address these design goals:

1. *sketching* 2D assets on a tablet
2. *positioning* these assets in space by positioning the tablet
3. *capturing* 3D spatial information along with a background video from a changing user viewpoint
4. *enacting* animations by physically moving the tablet

The third activity is key since the spatial information captured with the video lets the user later position and move the tablet in the captured area, enabling the positioning and enacting activities.

While any AR capable mobile device could work with PRONTO, using a tablet with a stylus makes sketching and video interactions more fluid.

An observational study with professional designers showed that PRONTO is easy to use, allowing them to create video prototypes quickly and effectively. Each designer prototyped both a predefined experience and an experience of their choosing. All the participants completed their prototypes more quickly and easily with PRONTO than they thought would be possible with their current tools.

RELATED WORK

While there are many rapid prototyping techniques for designing interactive systems [5], we focus on the ones used in, or most suitable for, AR.

Sketching for rapid prototyping

Sketch-based interactions have been widely explored in HCI for design and prototyping due to their central role in visual thinking, rapid prototyping, ideation, and communication.

Contextual Sketching

Contextual Sketching tools leverage the guidance provided by the real-world environment and underlying media, such as video and RGBD images. Previous 3D sketching systems use the existing visual media (3D scene geometry, models, and environment) to provide geometric constraints [2, 34] and aid 3D placement and snapping [28, 47]. PRONTO also leverages existing scene information, but instead of producing static 3D designs, our system creates a dynamic video that shows how graphical elements respond to user interactions in a 3D scene.

Interactive body-driven graphics [49] enable presenters to interact with graphical elements during live storytelling. However, unlike PRONTO, all the interactions are in 2D and are authored *before* the live presentation without any contextual information. PRONTO uses spatio-temporal synchronization to place and animate sketches in 3D, compositing them into a final video prototype.

Sketching Dynamic Concepts

Sketch-based interactions have been extensively explored in HCI for dynamic and animated content [10, 24, 29], facilitating rapid ideation and communication. In the context of prototyping and storyboarding, SILK [29] is one of the early explorations. It lets designers illustrate behaviors by sketching storyboards that specify how the screen should change in response to end-user actions. More recently, SketchStudio [26] enables designers to prototype, share, and review an animated design scenario involving complex design subjects. However,

³<https://www.torch.app/>

⁴https://www.youtube.com/watch?v=_3Y7BxEbqcg

⁵https://www.youtube.com/watch?v=BRIJG0x_We8

⁶<https://chalk.vuforia.com/>

the design and interaction of SketchStudio are different from ours, since it requires the designer to create a node graph of the user’s journey using a desktop environment, without information from a contextual scene.

Mobile and Physical Prototyping

Some prototyping tools rely on mobile devices not only as testing platforms but also as a design medium to author interaction. Tools like Enact [33] or Astral [31] rely on mobile devices to capture touch information and general sensor data, respectively. PRONTO also uses mobile devices as an active design tool, but unlike these prototyping environments, we designed PRONTO to support AR prototyping specifically.

Another way to explore AR environments is to create physical mockups and augment them with digital projections. For example, miniStudio [27] lets designers prototype ubiquitous computing spaces with a miniature physical model of the environment. However, creating these physical mockups requires preparation, and projections require a specialized prototyping environment, hindering the speed of the rapid prototyping process. PRONTO provides a zero installation environment and requires only one device, a tablet with a stylus.

Video Prototyping

Video prototyping [37, 38, 39] refers to creating video illustrations, generally using low-tech methods like paper prototyping [48] and the Wizard of Oz (WOz) technique [16]. A video prototype captures the user interaction to communicate ideas or to reflect on the design. These low-budget videos are well suited for early-stage design, but they can also be high-budget [51, 4] when a higher level of fidelity is needed. Both low-budget [53] and high-budget [52] video prototypes can illustrate AR designs. However, few tools support creating them.

Video prototyping systems

Remote Paper Prototype Testing, or RPPT [8], is used to run live testing sessions with real users. A “wizard” records a paper prototype with a fixed smartphone camera positioned above the surface. RPPT streams the video feed to the tester’s device with existing software, e.g., Google Hangouts. The tester interacts with this video feed as if it were an interactive UI. At the same time, the tester wears smart glasses with a camera, like Google Glass, that stream the user’s view and GPS location back to the wizard. With this information, the wizard gives instructions to the tester and updates the paper prototype accordingly. However, RPPT does not let designers iterate over the recorded video, i.e., RPPT targets testing rather than design.

Montage [32] is a video prototyping system that enables designers to progressively augment video prototypes with digital sketches, facilitating the creation, reuse, and exploration of dynamic interactions. It uses a combination of three mobile devices to livestream a paper prototype and a scenario to a tablet device with a stylus. Montage avoids the problems of live streaming in combination with the WOz technique by capturing the session and allowing designers to work on top of the recorded video afterward. Its iterative process is quite similar in spirit to PRONTO; however, Montage only works

in 2D, which limits the exploration of AR considerations and cannot address issues like 3D placement, spatial animation, and changes in user’s viewpoint.

AR prototyping systems

Many AR prototyping tools require extensive programming knowledge [23, 12]. Some, such as MARS [17], focus on one type of experience, in its case location-based AR. The mobile version of MARS [18] has a feature to author AR multimedia and hypermedia content, a technique called *Freeze-Frame*, that pauses tracking and lets the designer work in a single captured AR frame. However, MARS requires multiple devices, designers located both in- and outdoors, and focuses on anchored AR objects rather than dynamic experiences.

There are a few general non-specialized AR prototyping tools, including DART [36] and ComposAR [50]. However, they still require extensive coding or scripting skills to customize the experience. The DART system consists of a plugin for Macromedia Director that overlays graphics on top of the user’s view by using a see-through display. Designers can drag default behaviors onto the scene’s sprites, but new behaviors require coding.

DART allows users to embed animated assets into a video to illustrate an AR experience. However, users create the assets on a 2D desktop UI with indirect mouse and keyboard interactions. DART’s reflection paper states, “ARTIST expressed a desire to identify physical locations by moving through the space” [14], showing the need for a more direct approach. In contrast to DART, our system combines the authoring space and design space in a seamless workflow: users can author, animate, and embed assets to the video, all with a single AR device.

Combining AR tracking with other techniques

Prototyping tools can combine AR world-tracking with previous techniques, such as rapid prototyping, WOz simulations, and sketching.

ProtoAR [44] combines paper prototypes, in the form of sketches, and physical prototypes, in the form of clay models, with a comprehensive AR prototyping environment. The system is composed of a laptop with an authoring interface and a mobile phone to capture the paper prototypes and video snippets of the physical prototypes to illustrate a quasi-3D effect over the modeled object. These prototypes are positioned over fiducial images, i.e., markers, to support vision-based tracking. In contrast, PRONTO is a single device system that focuses on the prototyping of dynamic AR experiences, e.g., mid-air gestures and animations, rather than the creation of physical assets. Also, PRONTO simplifies 3D asset creation by allowing users to sketch out asset ideas using 2D sketches quickly. These sketches can be treated as replaceable placeholders for more polished 3D models later.

Some tools rely on WOz simulations, such as the already mentioned DART [11], WozARd [1], and 360Proto [43]. WozARd targets location-based AR experiences augmented with images, video, and sound. 360Proto combines paper prototyping with the WOz technique. This tool specializes in 360 experiences by creating paper prototypes of different spherical

layers surrounding the user. While the paper drawings are static, 360Proto supports live WOz for moving paper elements within the layers to simulate system responses. However, creating these paper layers is a laborious process; it is not easy to draw on a flat piece of paper that will be deformed to the surface of a sphere.

One of the main limitations of live WOz, shared by these systems, is designer fatigue—the wizard needs to react fast enough to the user inputs to convey the impression of interactivity. PRONTO also relies on demonstrations, which is similar to the WOz technique. However, these designer actions happen over a recorded video, similar to Montage [32], letting designers calmly iterate over the scenario and reducing fatigue compared with a live prototyping session.

Perhaps, the closest to our work is PintAR [15], which combines digital sketches on a tablet with live AR tracking from a HoloLens. However, it does not support the creation of custom animations and uses video just as an output of the design, not as a design medium in itself to reduce repetitive testing.

DESIGN GOALS

To inform the design of our system, we conducted three formative interviews with expert AR designers to identify current issues in early-stage AR design, using the critical incident technique [38]. We asked participants about their most recent AR projects and their main challenges for early-stage prototyping. Then we asked more general questions with a particular focus on understanding what they considered their most tedious activity. Four problems stood out.

Creating high-fidelity assets

All participants mentioned a vast ecosystem of tools to create 3D assets such as Blender, Maya, and Cinema4D. For example, P2 was exploring a design for a loader widget in AR. They created a high-fidelity asset in Cinema4D, later they exported it as an animated GIF, and then imported it into Keynote to explore different transition animations. However, all participants also reported using paper prototyping techniques during early-stage prototyping. To reduce the integration of many tools so early in the design process, and following the guidance of related work showing the value of sketches in prototypes (subsection 2.1), we want our tool to let designers sketch low-fidelity assets in context without sacrificing the ability to explore dynamic behaviors for them.

Handling the changing user position

Other challenges are a consequence of the freedom that the user has while using the system. For example, a user can walk around an object and interact from different points of view. P1 noted that this leads to a tedious “*reload everything, play, test*” cycle. P2 elaborated further: “*The focus is to stage a 2D presentation model of what would be an AR experience. We take what is the key moment and represent [it] as static stage. To add variations, we bring background into the scene and adjust it left and right between artboards, to illustrate different camera view.*” This finding resonates with what other designers express: “*One of the biggest challenges when designing for AR is your [the designer’s] inability to control the camera*” [41].

Positioning 3D assets

P1 and P2 said that the most tedious task is positioning 3D assets. To overcome this issue in early-stage prototyping, P1 relied on paper prototyping, body-storming, and acting-out the experience. For example, while designing a gaze-based AR interaction, they attached a real laser pointer to his head with tape to *feel* the experience of positioning a gaze marker. In that way, the laser pointer could represent the gaze marker, and it always followed the head’s orientation.

P2 mentioned that positioning is a problem, particularly with occlusion. They relied on rapid prototyping to illustrate an experience that required “*a user scanning the room.*” The final technology was not in place for this “*photogrammetry and 3D room scan*” system. Instead, P2 used photo-editing software with a camera preview image, positioning feature points to “*create noise*” and “*emulate what the point cloud looks like.*” P2’s driving questions were: “*what is the output [of the system]?*” and “*what is the experience?*”

Animations

All designers considered animation a pain point. P1 said they avoided laborious manual animation during early-stage prototyping: “*we do not do something crazy like keyframe animation, developer style.*” P2 mentioned problems with animating various objects and coordinating their interaction. For example, when discussing a rocket landing animation on a platform over the water, P2 said, “*we don’t want it to land through the platform to the ocean,*” and that for this, “*[designers] need a quick feedback loop for trial and error.*” P3 needed to create a path-based animation, and so created a mobile AR application to “draw” a path on a surface the app detected. Later, a high-fidelity 3D model, such as a dinosaur, was “connected” to the previously drawn path to create a moving animation.

Based on these interviews and the issues identified in the related work, we distilled four design goals for things a designer should be able to do in our new rapid prototyping tool:

- DG1: create low-fidelity assets
- DG2: capture user position changes
- DG3: position assets in space
- DG4: create complex animations

THE PRONTO SYSTEM

How can a rapid prototyping system fulfill our goals?

Design rationale

The most popular platforms for AR today are mobile devices, specifically smartphones, and tablets. Users generally hold the device with one or two hands while interacting with the system by moving the device in the scene or by making one-handed gestures, either mid-air or over the screen. AR headsets allow bi-manual interaction and are becoming more reliable and available every day, but we wanted to make a tool accessible to as many stakeholders as possible. While mobile devices have interaction limitations and are less powerful than headsets, their ubiquity provides a lower barrier of entry. They also simplify authoring because they give designers a physical surface to interact with, and because 2D interfaces are more familiar

than headset interfaces. For these reasons, we implemented PRONTO on a mobile tablet, which provides more screen real estate and better stylus support than a smartphone.

Sketching to create low-fidelity assets (DG1)

Designers like to use quick sketching for early-stage design [7]. However, for AR design, they usually need complex software to create and import 3D assets. We want to allow designers to sketch placeholders for future high-fidelity 3D assets without incurring the cost of creating those assets too early in the design process. Simple 2D sketches positioned in the 3D scene are sufficient to convey early design ideas [6]. PRONTO allows designers to use the full tablet screen and a stylus to sketch assets.

Video to capture the user position changes (DG2)

Creating an interactive system involves many repetitive and redundant activities [40]. For interactive AR prototypes, the designer needs to replicate the particular user scenario every time the AR interface changes, i.e., to arrange the real objects, surfaces, and lighting conditions of the scene and play the role of the user—moving, touching, and speaking—on every test run. To minimize repetition, we want to fix, or lock, the use scenario and let the designer iterate over the AR interface. Video prototypes are an excellent way to achieve this. Although they are non-interactive, their playback capabilities illustrate dynamic AR interfaces over a recorded scenario. In PRONTO, designers begin by capturing a video showing a changing user viewpoint of the environment. Then they use a familiar user interface to play the video back and to set times for overlaid interface responses.

Direct manipulation for placement (DG3) and animation (DG4)

Paper-based prototyping techniques have the advantage of being direct and simple to use. Designers generally use the WOZ technique to interact with paper mockups directly. In the same way, PRONTO supports placing and animating 3D objects by manipulating them in their real environment. Positioning 3D assets on an AR experience should be as easy as placing a cup of coffee on top of a table. By capturing spatial information at the same time it captures the background video, PRONTO provides *on-video 3D navigation* of the tracked space to achieve this without markers, specialized tracking rooms, or multiple devices. To place an asset, designers simply position the tablet where the asset should go (Figure 1). Similarly, they can create animations by demonstration, *enacting* the movement of objects with the tablet. The system captures the movement to create the desired animation.

Interface and workflow

PRONTO has a simple interface. The first screen of the system lets designers record the background video. The main features are on the second screen, i.e., the playback screen, shown in Figure 2. This view has three main UI components: a playback timeline (Figure 2A) to manipulate the recorded video, a representation of the current tablet's position (Figure 2B), and spatial layers with sketches (Figure 2C). Other labeled components in the figure are explained later in the section.

A concrete example is the best way to explain the main features of PRONTO. The designer wants to prototype an AR shopping



Figure 2. PRONTO's playback screen: (A) The timeline can play/pause the recorded video shown in the background, (B) a visualization of the current tablet's position, (C) a spatial layer with a sketch, (D) the "Sketch" button creates new spatial layers at the tablet's position, (E) the "Grab-Drop" moves the selected spatial layer by moving the tablet, (F) the selected layer can appear/disappear/face the camera or be deleted, (G) the timeline shows when the selected spatial layer appears/disappears, (H) and the selected animation beginning and duration, (I) undo and stroke color.

experience. The final system will run on a headset capable of detecting physical hand gestures like tapping on the table or pinching mid-air. The planned interface has the user tapping on a product to display a virtual price tag above the product, and pinching the tag to buy the product, which will make the tag fly out of the scene. The designer uses PRONTO to produce a video that captures the entire shopping experience, including the interactions and their results.

Step 1: Acting and Capturing

The first step is to capture a video using the tablet's camera. The designer starts by pressing the "Record" button. The video captures the physical environment and includes the product, e.g., a coffee mug, that a user might buy. During recording, the designer acts out several interactions using the dominant hand while holding the tablet in the other hand. They tap the product, which will pop up a virtual price tag, then they make a pinch gesture, which will make the tag fly out of the scene along a predefined 3D path. The designer also moves around to change viewpoints during the recording. Pressing the "Stop" button finishes the capture.

Unlike existing video prototyping tools, PRONTO stores both the recorded video and synchronized 3D scene information, including camera position and orientation and detected surfaces and geometry. We call this capture an *augmented video recording*. The 3D scene information enables placing and animating graphical elements on the video, similar to a live AR experience.

Step 2: Creating a Spatial Layer for an Asset

After recording, the designer can rewind the recorded video to add the price tag, placing it as a video overlay above the product. The designer first scrubs the timeline (Figure 2A) to the point where tap gesture started. Whenever the video is paused,

PRONTO tracks the tablet in the real-world environment using the tablet’s camera and the data from the augmented video recording. The system overlays a tablet visualization on the video frame to show the current position and orientation (Figure 2B). Once the tablet is where the asset should appear, i.e., over the product, the designer presses the “Sketch” button (Figure 2D). This creates a *spatial layer* at the current location (Figure 2C), which acts as a virtual 2D canvas for sketching. The designer then sketches a price tag on the layer using a stylus.

The designer can set the appearance and disappearance time for a spatial layer by pressing the open and closed eye buttons (Figure 2F) at the appropriate times. In this case, the price tag layer should appear at the tap and disappear shortly after the pinch; these times are marked in the timeline with the same eye icons (Figure 2G). The user can playback the prototype so far. The drawing of the price tag will remain in the same spatial location above the product, even as the viewpoint changes in the video.

Step 3: Enacting an Animation

In this final step, the designer performs the fly-out animation in response to the pinch gesture. The designer scrubs the timeline to the pinch gesture and moves the tablet so that it intersects the price tag drawing; the tablet image overlaid in the video makes this positioning easy. Then, they press the “Enact” button (Figure 2D) and perform the animation by moving the tablet away from the product, following any desired trajectory. The 6DOF position and orientation of the tablet’s movement are recorded as a motion path and applied to the price tag, making it fly out of the scene. This *enaction* through direct manipulation lets the designer express a complex 3D motion trajectory intuitively. The final augmented video is included in the supplemental material.

This demonstrates an end-to-end workflow for creating the AR experience video. PRONTO leverages familiar 2D interactions like sketching and scrubbing a timeline, and 3D spatial interactions with 6DOF, to composite graphical elements in an augmented video. Using our novel augmented video recording to guide the creation, 3D positioning, and animation of spatial layers is the defining aspect of PRONTO. In the following sub-sections, we discuss the creation and animation of spatial layers in more detail.

Augmented Video Recording

The augmented video recording is a key contribution of PRONTO. During the capture, we record the tablet’s position and orientation in space at each frame. The result is a 6DOF camera trajectory through space. This, combined with information about the camera optics, lets PRONTO render placed objects to match the environment video.

We also record all of the environment information provided by the AR SDK, including detected surfaces and feature points on objects. Although we do not use this information in the current version, it enables planned future enhancements like using surfaces to help place and crop assets.

Spatial Layers: Creation and Manipulation

Being able to define, place, and choose sketch planes quickly is critical to 3D sketching systems [3, 2]. In PRONTO, sketch planes are called spatial layers. When the video is paused, PRONTO renders a spatial layer as a transparent blue grid, and when the video is playing, the grid disappears, showing only the sketches the designer had drawn on the spatial layer.

To create a spatial layer, the designer positions the tablet in space and taps the “Sketch” button, at which time a grid appears on the screen of the tablet image that is being composited into the scene. This grid remains in place when the designer moves the tablet away, and the designer can scrub the timeline to see the grid from different viewpoints.

Non-trivial prototypes usually contain multiple spatial layers. The designer can directly select a particular spatial layer by tapping or indirectly from a list of spatial layers on the right panel. The latter is especially useful in dense prototypes with many spatial layers. Only one spatial layer can be selected at a time.

There are several things the designer can do with a selected spatial layer.

2D Affine Transformations

The spatial layer is initially the size of the tablet screen. The designer can uniformly scale, move, and rotate it using a standard set of multi-touch gestures. For simplicity, these transformations are limited to the spatial layer’s plane. Since they are affine, the spatial layer always remains a rectangle.

Grab and drop

The designer can make arbitrary changes to a spatial layer’s position and orientation by touching and holding the “Grab” button in the user interface and moving the tablet. The selected layer moves relative to the tablet, and releasing the button drops the selected layer in the new position. Layers can be grabbed from a distance, which allows them to be positioned and animated in spaces beyond the designer’s reach. Note that “grab and drop” is the only way to statically move or rotate a spatial layer out of its plane.

In PRONTO, the direct mapping between the virtual and physical spaces facilitates direct manipulation interactions for positioning and animation. As illustrated in Figure 1: to add an asset to the table in the video, the user places the tablet on the actual table in the room.

Sketching

To draw on a spatial layer, the designer can draw on top of it with the stylus or double-tap it to go full-screen. In full-screen, the spatial layer’s grid expands to fill the tablet screen, backed by a white background, and the designer can sketch on the spatial layer with a stylus. PRONTO provides usual sketching tools and widgets like a color picker, stroke width slider and redo/undo, and clear (only available in full-screen). Double tapping the full-screen view switches back to the default playback view, where the spatial layer’s sketch is drawn on the grid when the video is paused, this grid is hidden when the video is playing.

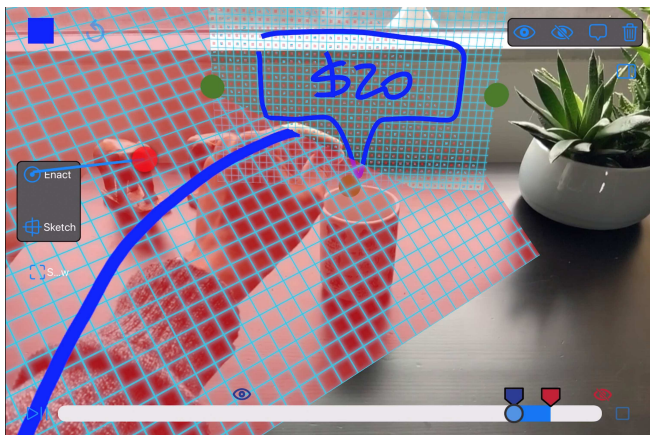


Figure 3. Indirect enaction: to animate the price tag according to the sketched motion-path the designer: (1) creates a new spatial layer with the motion-path (2) selects the spatial layer with the price tag (3) creates a link from the “Enact” button to the motion-path spatial layer.

Animation

Besides the static actions described in the previous section, the user can animate the spatial layer in four ways.

Direct enaction

This is the simplest animation technique. The designer taps the “Enact” button and then holds the “Grab-Drop” button (Figure 2E) while “grabbing” the selected spatial layer with the tablet device and moving it in 3D space to record an animation. The spatial layer follows the 6DOF position and orientation of the tablet, but applied relative to the current 3D transformation of the spatial layer, preserving its initial position and orientation.

Indirect enaction

Sometimes direct enaction is impractical or difficult; for example, the designer might want the spatial layer to move in a straight line or follow a specific path. Experience with VR drawing programs have shown that freehand drawing in space is easy, but drawing a specifically-desired path in space is very difficult.

To support this, the designer can create a second spatial layer and draw a motion path on it. For example, the designer could make the price tag move directly towards the camera, simulating a purchasing action, by positioning the tablet perpendicular to the sketch, creating a new spatial layer, resizing it, and drawing a motion path (Figure 3). Rather than using the “Grab-Drop” button to animate the selected spatial layer, as is done with direct enaction, the designer uses the stylus to create a link, dragging a line from the “Enact” button to the spatial layer with the motion path, and releasing. This attaches the selected spatial layer to that path. The motion-path spatial layers can be hidden with the disappear button at the beginning of the video. In a future version, these motion-path spatial layers will hide automatically during playback.

Because spatial layers are planar, indirect enaction only supports 2D motion paths.

Cropping

Many experiences involve having an asset appear or disappear progressively. To support this, when a spatial layer is selected, four cropping handles appear on the edges of the grid—the four green dots in Figure 2C. The designer can drag the edge to progressively crop the spatial layer’s sketch, or drag it back to its original position to reveal it progressively. In order to create an animation, the designer presses the “Enact” button and drags a cropping handle. PRONTO creates the demonstrated cropping animation with that action duration.

Visibility and timing

When a spatial layer is selected, eye icons above the timeline show when it becomes visible (Figure 2G), and when it disappears. By default, the spatial layer becomes visible at the time it was placed and remains visible until the end of the video. To change this, a designer can scrub the timeline to the desired time and tap the open or closed eye buttons (Figure 2F).

Animations also have start and end markers in the timeline if the spatial layer that they animate is selected. An *animation bar* appears between them (Figure 2H). Designers can move an animation in time by dragging its animation bar as a whole, or stretch/shorten it by dragging the start/end marker. This design is familiar and simple but is effective in fine-tuning the timing and coordinating multiple animations.

Implementation

Our current version is implemented on an iPad Pro 2 with an Apple pencil as the stylus. The augmented video recording is captured using Apple’s ARKit⁷ and rendered with SceneKit⁸.

When the user records the video, we intercept ARKit’s rendering to extract the inferred AR information, including the transformation matrix of the recording tablet’s inferred position and orientation at every frame. We generate a movie file (mp4) from each frame’s pixel buffer and a time-coded data file (JSON) that contains the AR information.

The playback view shows a SceneKit 3D scene controlled with a timeline slider and a play/pause button. The scene has a 2D background, a camera, and the sketched layers. The background is a texture playing the recorded movie file, synchronized with the timeline slider. The camera’s properties, such as focal length, are set to match the properties of the recording tablet’s actual camera. The sketched layers have possibly-animated positions and orientations as set by the designer.

Whenever the scene time changes, either because of the playback or because the designer scrubs the timeline slider, we retrieve the AR information for the frame of the recorded video that is closest in time. We then set the scene’s camera to match the position and orientation of the tablet during that frame. This makes the sketched layers seamlessly match the background.

⁷<https://developer.apple.com/documentation/arkit>

⁸<https://developer.apple.com/documentation/scenekit>

AR EXPERT REVIEW STUDY

We conducted an expert review study with AR professionals to evaluate PRONTO. Our goals were to 1) observe and assess how expert users use PRONTO to prototype AR and 2) examine the qualitative usability and utility of PRONTO. Our evaluation strategy falls into the category of “usage evaluation” as described in the HCI toolkit evaluation strategy classification [30].

A more controlled experiment to compare PRONTO with a baseline would be difficult because there is no clear baseline. Current video-based AR prototyping is most often done with professional editing tools like After Effects. This software is designed for a different purpose, high fidelity output, and use very different technologies to work with conventional video using a desktop interface. We also did not test PRONTO on novice users, relying on experts to gain insights from their workflows. We expect that this initial feedback will help distill the strengths and areas for improvement of PRONTO for future iteration.

We recruited six AR professionals ($n = 6$, two female). Participants self-identified as Design Consultant (P1), Experience Developer (P2 & P3), Experience Designer (P4 & P5), and animation designer (P6). Their ages ranged from 23 to 44. We recruited participants based on their design seniority. Their years of general design experience ranged from 2 to 20 years ($M = 12, SD = 7.04$). All the participants had done at least one AR project. Participants previously worked in one to five AR projects ($M = 2.33, SD = 1.51$). We also asked participants what tools they use for AR prototyping (Figure 4).

Procedure

Participants were first introduced to PRONTO and then prototyped two AR experiences: one implementing a predefined experience and the other open-ended. They then completed questionnaires comparing PRONTO to their workflows, and evaluating PRONTO’s two main features, asset creation and animation. Our questions were inspired by the “first-use study” in Exemplar [19].

Introduction and training (15 minutes)

Upon arrival, participants were introduced to all of the features of the PRONTO system. They were then guided through creating a short AR prototype in which tapping on a table made an animated panel appear.

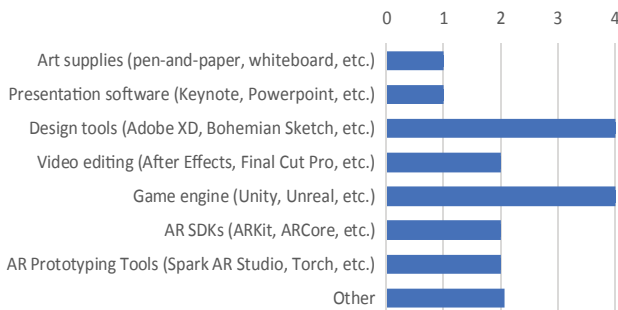


Figure 4. Participants’ preferences of AR prototyping tools.

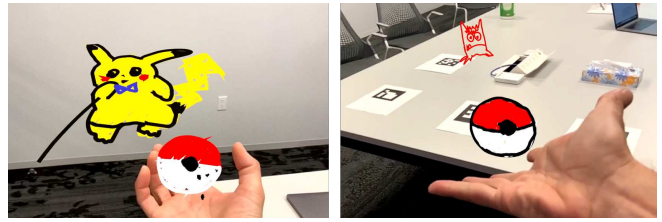


Figure 5. Sample outputs of Task 1—Pokemon AR (from P4 and P5).

Task 1: Pokemon AR (15 minutes)

In this task, participants prototyped a new catch mechanism in Pokemon Go AR, following specific instructions. They were shown a video of Pokemon Go and were asked to replace the touch gestures in Pokemon Go with hand gestures. The instructions were as follows: 1) Add a Pokemon to the scene, 2) Add a Pokeball on your hand, 3) Make the Pokeball hit the Pokemon when you flick it, 4) Make the Pokemon shake and disappear when it gets hit. Participants were given a maximum of 10 minutes to complete this task and were asked to complete it as fast as possible.

Task 2: Open-ended (30 minutes)

Participants were then asked to describe an AR experience that they would like to create, and were asked to use PRONTO to prototype it. Clarifications were provided if they asked questions about the system.

After all the tasks are done, participants filled out the questionnaires. We also asked them for additional comments and feedback. The study lasted about 60 minutes, and we compensated the participants for their time.

RESULTS AND DISCUSSION

Table 1 shows a summary of participants’ completion time and task outputs. All participants completed both tasks. In Task 1 (Pokemon), all participants except P4 and P6 finished in approximately 10 minutes. P4 was personally interested in the task and requested extra time to polish the sketch prototype (Figure 5 Left). P6 also requested more time to add an animation indicating the Pokemon frowns before disappearing. In Task 2 (Open-ended), participants prototyped six unique AR experiences. P1 created a narrative bedtime AR story for children. P2 created a game in which users view vehicles racing on the floor of a meeting room. P3 explored the mechanics of using a finger to tap AR buttons. P4 and P5 created instructional experiences explaining how to use physical products in AR. Finally, P6 created a first-person shooter (FPS) game concept in which a user fires with a hand gesture.

Strengths

PRONTO enabled participants to explore a wide range of advanced AR interface mechanics and interaction systems, many of which are normally difficult to prototype. For example, P1 created an offscreen “attention-guiding” cue [21] at the beginning of the prototype to guide viewers toward where the AR story is happening. The AR gesture interaction in P3’s prototype was roughly similar to those in a world-overlay interface from a complex projection system [55]. P4 and P5

Participants	Task 1 Pokemon (minutes)	Task 2 Open-ended (minutes)	Task 2 description & duration (seconds)
P1	7:07	18:31	Narrative-based experience (7s)
P2	7:54	11:07	Racing game (14s)
P3	10:16	8:49	Gesture-based interface with two buttons (13s)
P4	18:41	10:57	Product tutorial (27s)
P5	9:55	10:25	Smart home experience (31s)
P6	15:02	18:34	Gesture-based FPS game concept (18s)



Table 1. Results: a summary of task completion times (Task 1 and 2) and the resulting output (Task 2), i.e., AR video prototype description and duration

demonstrated AR experiences in which virtual graphics and annotations responded to user interactions with physical objects; these interactions often require complex sensing and user tracking technologies in previous AR tutorial research [20, 42]. P6 leveraged the camera movement to prototype the mechanics of an FPS gesture-based game. Our participants successfully recreated these advanced AR mechanics using PRONTO in only a short amount of time—the average completion time of Task 2 was 13:03 minutes. When asked to evaluate whether PRONTO encouraged exploration and creativity (Figure 6 Q3, Q4), 4/6 participants ranked it highly.

Figure 6 shows an overview of participants’ ratings of other aspects of PRONTO. The system was considered fun and easy to use. All experts ranked it highly for the statement “allows people with no AR knowledge to participate” (Q5).

In Q6 and Q7, most participants (6/6 for Q6 and 4/6 for Q7) agreed that that PRONTO helped them focus on the AR user and the interactions. A key aspect of PRONTO is the recording of video before prototyping. This unique “capture first, prototype later” workflow requires the designers to anticipate dynamic and user-oriented design elements like the user’s potential camera paths and interactions. This capture process helps put the designer directly into the AR user point of view. P5 explained the main benefit of PRONTO as, “Immediately get you into a headspace for thinking of AR instead of having to extrapolate on a desktop application.”. P6 shared similar remarks, “Desktop is nice but doing in AR is more helpful if you design on AR. It’s like artists who work on big screen but design for small screen. It’s easy to distance yourself from the final content.”

Several participants also agreed that PRONTO helped decrease time to materialize and explore AR designs (3/6 in Q1 and Q2). P6 mentioned that being able to sketch graphics and animation rapidly is a major advantage, explaining, “When you work in animation, you start realizing that you need to rely on other 3D artists or rigging artists to hand you the model. Here I can just draw it out.” P6 also explained that being able to combine animations with the video playback enabled her to experiment with causal effects (i.e., if this button is pressed, then this event happens). Normally, they would have to wait for an engineer

to programmatically build an event tree before being able to design on it.

In summary, the feedback from our expert review study revealed two key benefits of PRONTO. First, it encouraged designers to explore more creative and interactive designs in AR even if they had limited knowledge about the technicality of AR. Second, the unique “capture first, prototype later” workflow—the combination of video recording and in-situ AR creation tools—not only helped designers create ideas directly in AR but also helped them focus more on the user and the potential interactions.

Areas for improvement

We also collected feedback from participants who were neutral or in disagreement with survey questions to identify areas for improvement.

Several participants suggested that better tools to manage spatial layers could help improve “time to materialize the design” (Q1) and “rapid modification” (Q8). Participants reported that it was sometimes difficult to locate the spatial layer in the video. P4 explained, “You have to walk back to the exact location the shot was filmed.” P5 added, “Sometimes I don’t realize my graphics or animation will be out of frame.” Both participants suggested that directional cues like arrows or minimaps could help.

All prototypes created by our participants were rather short, ranging from 7 to 31 seconds (Table 1). P6 added that for bigger projects with more spatial layers, tools to organize the spatial layers efficiently in space and on the timeline would be necessary.

The animation workflow also received mixed feedback from the participants (Q10). They thought it would not scale well for longer actions. Currently, the video remains paused while enacting an animation. Both P2 and P5 reported that it was difficult to map what they were enacting with what would happen in the video. For example, P2 wanted the vehicle in their prototype to speed up, slow down, and turn at different times in the video. P6 shared similar concerns but emphasized that the simplicity of the workflow was what they liked about it. Solutions to improve mapping animation to the video could include better keyframe interfaces and multiple animation

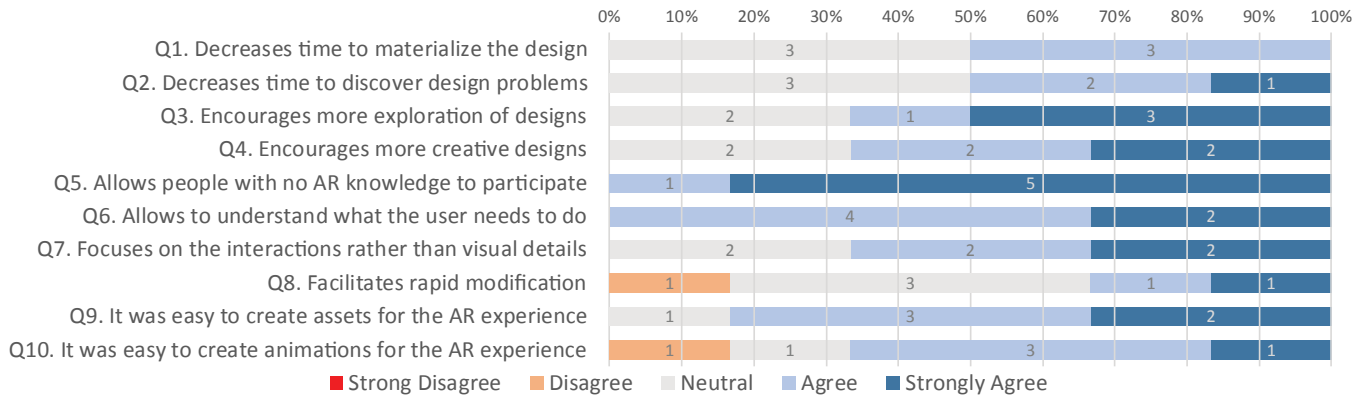


Figure 6. Results: 100% stacked bar chart ($n = 6$) of participants' ratings on "Do you agree or disagree with the following statements when you compare PRONTO with your prior practices?" (Q1-Q8) and "Do you agree or disagree with the following statements about PRONTO?" (Q9 and Q10).

tracks. However, they both would add significant complexity to PRONTO's interface and workflow, and therefore would need careful future investigation.

Limitations

During the study, participants were able to prototype many behaviors, including multi-object animation (P1, P2), gestural interactions (P3, P6), and instructions (P4, P5). However, our simple interface is not optimized for objects with several states, requiring a layer for each state. Making an object appear multiple times also requires layer duplication.

Our study participants did not find the "capture first, prototype later" workflow to constrain their design processes. Because they were in the physical space of the experience they wanted to prototype, they could rehearse and use props, and could quickly re-capture a new video if desired. However, because our system only supports a single video, it cannot easily prototype a changing environment or capture multiple viewpoints.

CONCLUSION AND FUTURE WORK

We presented PRONTO, a rapid prototyping tool for AR experiences that uses augmented video, sketching, and enaction using a tablet. We interviewed three AR expert designers and formulated a set of design goals to address the current limitations of AR prototyping tools. Based on these goals, we have demonstrated an end-to-end workflow and interaction technique to create and modify AR objects on an augmented recorded video. Our work expands the body of HCI research in prototyping tools for AR by combining sketching, enaction, and augmented video. Finally, we evaluated the system with expert AR designers, showing the potential of the workflow for rapid prototyping in contrast to current practices.

Our work suggests several interesting directions for future research. We would like to explore enhanced capturing capabilities to facilitate the creation of interaction during playback. To this end, we would like to employ state-of-the-art computer vision and sensing capabilities to capture user actions—hand gestures, gaze, and skeleton tracking—and environment information, such as a 3D depth map. This would significantly

enhance the drawing and animation workflow for better spatio-temporal anchoring. For instance, the designer can sketch an object and attach it to the user's hand in the recorded video. We would also like to enhance the capture with audio to prototype voice interaction.

Timing animations and coordinating them with the changing user viewpoint is a weakness of our current system. We would like to investigate enhancing this while still maintaining the simplicity of a single timeline. We would also like to explore incorporating multiple video captures to accommodate changing environments or multiple viewpoints.

Although our current workflow requires the designer to be in the same space for capturing and prototyping, in the future, we would like to explore remote capturing, working with scales spanning from the miniature to the massive, e.g., underwater or a drone-captured urban space. This would broaden the scope and capabilities of PRONTO beyond prototyping, to the design and analysis in engineering and architecture.

We would also like to enhance the expressive qualities of animations by drawing insights from the rich literature of performance and sketch-based animation tools in HCI and graphics [25, 54]. Beyond animations, supporting annotations, commenting, reviewing, recapturing the video without losing the sketches, and collaboration would significantly enhance the capabilities of PRONTO. Validating these ideas is an exciting future work direction that would further expand the scope of contextual prototyping and ideation.

ACKNOWLEDGMENTS

This research began during an internship at Adobe Research and was sponsored in part by Adobe Research. We would also like to thank Koji Yatani for hosting Germán at the University of Tokyo while writing part of this manuscript.

REFERENCES

- [1] Günter Alce, Mattias Wallergård, and Klas Hermodsson. 2015. WozARd: A Wizard of Oz Method for Wearable Augmented Reality Interaction – A Pilot Study. *Advances in Human-Computer Interaction 2015* (jun

- 2015), 1–10. DOI :
<http://dx.doi.org/10.1155/2015/271231>
- [2] 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. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '18)* (2018). DOI :
<http://dx.doi.org/10.1145/3173574.3173759>
- [3] Seok-Hyung Bae, Ravin Balakrishnan, and Karan Singh. 2009. EverybodyLovesSketch: 3D Sketching for a Broader Audience. In *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, New York, NY, USA, 59–68. DOI :<http://dx.doi.org/10.1145/1622176.1622189>
- [4] Jakob Bardram, Claus Bossen, Andreas Lykke-Olesen, Rune Nielsen, and Kim Halskov Madsen. 2002. Virtual video prototyping of pervasive healthcare systems. In *Proceedings of the conference on Designing interactive systems processes, practices, methods, and techniques - DIS '02*. 167. DOI :
<http://dx.doi.org/10.1145/778712.778738>
- [5] Michel Beaudouin-Lafon and Wendy E. Mackay. 2003. Prototyping tools and techniques. In *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*. 1017–1039. DOI :
<http://dx.doi.org/10.1201/9781410615862>
- [6] Bill Buxton. 2007. *Sketching User Experiences: getting the design right and the right design*. Morgan Kaufmann. 448 pages. DOI :
<http://dx.doi.org/10.1016/B978-0-12-374037-3.X5043-3>
- [7] Adam S. Carter and Christopher D. Hundhausen. 2010. How is User Interface Prototyping Really Done in Practice? A Survey of User Interface Designers. In *2010 IEEE Symposium on Visual Languages and Human-Centric Computing*. IEEE, 207–211. DOI :
<http://dx.doi.org/10.1109/VLHCC.2010.36>
- [8] Kevin Chen and Haoqi Zhang. 2015. Remote Paper Prototype Testing. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. ACM Press, New York, New York, USA, 77–80. DOI :
<http://dx.doi.org/10.1145/2702123.2702423>
- [9] Crunchbase. 2019. List of top Augmented Reality Companies. (2019). <https://www.crunchbase.com/hub/augmented-reality-companies>
- [10] Richard C. Davis, Brien Colwell, and James A Landay. 2008. K-Sketch : A “Kinetic” Sketch Pad for Novice Animators. (2008), 413–422.
- [11] Steven Dow, Jaemin Lee, Christopher Oezbek, Blair MacIntyre, Jay David Bolter, and Maribeth Gandy. 2005. Wizard of Oz interfaces for mixed reality applications. In *CHI '05 extended abstracts on Human factors in computing systems - CHI '05*. ACM Press, New York, New York, USA, 1339. DOI :
<http://dx.doi.org/10.1145/1056808.1056911>
- [12] M Fiala. 2005. ARTag, a fiducial marker system using digital techniques. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 2. 590–596 vol. 2. DOI :
<http://dx.doi.org/10.1109/CVPR.2005.74>
- [13] Interaction Design Foundation. 2019. Augmented Reality – The Past, The Present and The Future. (2019). <https://www.interaction-design.org/literature/article/augmented-reality-the-past-the-present-and-the-future>
- [14] Maribeth Gandy and Blair MacIntyre. 2014. Designer’s augmented reality toolkit, ten years later. In *Proceedings of the 27th annual ACM symposium on User interface software and technology - UIST '14*. ACM Press, New York, New York, USA, 627–636. DOI :
<http://dx.doi.org/10.1145/2642918.2647369>
- [15] 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 (CHI EA '19)*. ACM, New York, NY, USA, LBW1416:1—LBW1416:6. DOI :
<http://dx.doi.org/10.1145/3290607.3312847>
- [16] Paul Green and Lisa Wei-Haas. 1985. The Rapid Development of User Interfaces: Experience with the Wizard of OZ Method. *Proceedings of the Human Factors Society Annual Meeting 29, 5* (1985), 470–474. DOI :<http://dx.doi.org/10.1177/154193128502900515>
- [17] S Guven and S Feiner. 2003. Authoring 3D hypermedia for wearable augmented and virtual reality. In *Seventh IEEE International Symposium on Wearable Computers, 2003. Proceedings*. 118–126. DOI :
<http://dx.doi.org/10.1109/ISWC.2003.1241401>
- [18] Sinem Guven, Steven Feiner, and Ohan Oda. 2006. Mobile augmented reality interaction techniques for authoring situated media on-site. In *2006 IEEE/ACM International Symposium on Mixed and Augmented Reality*. IEEE, 235–236. DOI :
<http://dx.doi.org/10.1109/ISMAR.2006.297821>
- [19] Björn Hartmann, Leith Abdulla, Manas Mittal, and Scott R. Klemmer. 2007. Authoring sensor-based interactions by demonstration with direct manipulation and pattern recognition. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07*. ACM Press, New York, New York, USA, 145. DOI :<http://dx.doi.org/10.1145/1240624.1240646>
- [20] Steven Henderson and Steven Feiner. 2010. Exploring the benefits of augmented reality documentation for maintenance and repair. *IEEE transactions on*

- visualization and computer graphics* 17, 10 (2010), 1355–1368.
- [21] Steven J Henderson and Steven Feiner. 2009. Evaluating the benefits of augmented reality for task localization in maintenance of an armored personnel carrier turret. In *2009 8th IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 135–144.
- [22] David G. Jansson and Steven M. Smith. 1991. Design fixation. *Design Studies* 12, 1 (1991), 3–11. DOI: [http://dx.doi.org/https://doi.org/10.1016/0142-694X\(91\)90003-F](http://dx.doi.org/https://doi.org/10.1016/0142-694X(91)90003-F)
- [23] H Kato and M Billinghamurst. 1999. Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In *Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99)*. 85–94. DOI: <http://dx.doi.org/10.1109/IWAR.1999.803809>
- [24] Rubaiat Habib Kazi, Fanny Chevalier, Tovi Grossman, and George Fitzmaurice. 2014. Kitty: Sketching Dynamic and Interactive Illustrations. In *Proceedings of the 27th annual ACM symposium on User interface software and technology - UIST '14*. ACM Press, New York, New York, USA, 395–405. DOI: <http://dx.doi.org/10.1145/2642918.2647375>
- [25] Rubaiat Habib Kazi, Tovi Grossman, Nobuyuki Umetani, and George Fitzmaurice. 2016. Motion Amplifiers: Sketching Dynamic Illustrations Using the Principles of 2D Animation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 4599–4609. DOI: <http://dx.doi.org/10.1145/2858036.2858386>
- [26] Han-Jong Kim, Chang Min Kim, and Tek-Jin Nam. 2018. SketchStudio: Experience Prototyping with 2.5-Dimensional Animated Design Scenarios. *Proceedings of the 2018 on Designing Interactive Systems Conference 2018 (2018)*, 831–843. DOI: <http://dx.doi.org/10.1145/3196709.3196736>
- [27] Han-Jong Kim, Ju-Whan Kim, and Tek-Jin Nam. 2016. miniStudio: Designers' Tool for Prototyping Ubicomp Space with Interactive Miniature. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*. ACM Press, New York, New York, USA, 213–224. DOI: <http://dx.doi.org/10.1145/2858036.2858180>
- [28] 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 (CHI '19)*. ACM, New York, NY, USA, 176:1—176:11. DOI: <http://dx.doi.org/10.1145/3290605.3300406>
- [29] James A. Landay and Brad A. Myers. 1995. Interactive Sketching for the Early Stages of User Interface Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '95)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 43–50. DOI: <http://dx.doi.org/10.1145/223904.223910>
- [30] David Ledo, Steven Houben, Jo Vermeulen, Nicolai Marquardt, Lora Oehlberg, and Saul Greenberg. 2018. Evaluation Strategies for HCI Toolkit Research. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*. DOI: <http://dx.doi.org/10.1145/3173574.3173610>
- [31] David Ledo, Jo Vermeulen, Sheelagh Carpendale, Saul Greenberg, Lora Oehlberg, and Sebastian Boring. 2019. Astral: Prototyping Mobile and Smart Object Interactive Behaviours Using Familiar Applications. In *Proceedings of the 2019 on Designing Interactive Systems Conference (DIS '19)*. ACM, New York, NY, USA, 711–724. DOI: <http://dx.doi.org/10.1145/3322276.3322329>
- [32] Germán Leiva and Michel Beaudouin-Lafon. 2018. Montage: A Video Prototyping System to Reduce Re-Shooting and Increase Re-Usability. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology - UIST '18*. ACM Press, Berlin, Germany. DOI: <http://dx.doi.org/10.1145/3242587.3242613>
- [33] Germán Leiva, Nolwenn Maudet, Wendy Mackay, and Michel Beaudouin-Lafon. 2019. Enact: Reducing Designer-Developer Breakdowns When Prototyping Custom Interactions. *ACM Trans. Comput.-Hum. Interact.* 26, 3 (may 2019), 19:1—19:48. DOI: <http://dx.doi.org/10.1145/3310276>
- [34] Yuwei Li, Xi Luo, Youyi Zheng, Pengfei Xu, and Hongbo Fu. 2017. SweepCanvas: Sketch-based 3D Prototyping on an RGB-D Image. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. ACM, New York, NY, USA, 387–399. DOI: <http://dx.doi.org/10.1145/3126594.3126611>
- [35] Danwei Tran Luciani and Peter Vistisen. 2017. Empowering Non-Designers Through Animation-based Sketching. 7 (2017). <http://www.nordes.org/nordes2017/assets/full>
- [36] Blair MacIntyre, Maribeth Gandy, Steven Dow, and Jay David Bolter. 2004. DART: A Toolkit for Rapid Design Exploration of Augmented Reality Experiences Blair. In *Proceedings of the 17th annual ACM symposium on User interface software and technology - UIST '04*. ACM Press, New York, New York, USA, 197. DOI: <http://dx.doi.org/10.1145/1029632.1029669>
- [37] Wendy E. Mackay. 1988. *Video Prototyping: a technique for developing hypermedia systems*. Vol. 5. ACM/SIGCHI.
- [38] Wendy E. Mackay. 2002. *Using video to support interaction design*. INRIA Multimedia Services. <https://www.lri.fr/>

- [39] Wendy E. Mackay and Anne-Laure Fayard. 1999. Video brainstorming and prototyping: techniques for participatory design. *CHI'99 extended abstracts on Human factors in ...* May (1999), 118–119. DOI: <http://dx.doi.org/10.1145/632716.632790>
- [40] Nolwenn Maudet, Germán Leiva, Michel Beaudouin-Lafon, and Wendy E. Mackay. 2017. Design Breakdowns: Designer-Developer Gaps in Representing and Interpreting Interactive Systems. *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing - CSCW '17* (2017), 630–641. DOI: <http://dx.doi.org/10.1145/2998181.2998190>
- [41] Michael Flarup. 2018. Designing For Augmented Reality. (2018). <https://www.youtube.com/watch?v=-r1VtYxvVIc>
- [42] Peter Mohr, David Mandl, Markus Tatzgern, Eduardo Veas, Dieter Schmalstieg, and Denis Kalkofen. 2017. Retargeting video tutorials showing tools with surface contact to augmented reality. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 6547–6558.
- [43] Michael Nebeling and Katy Madier. 2019. 360Proto: Making Interactive Virtual Reality & Augmented Reality Prototypes from Paper. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, 596:1—596:13. DOI: <http://dx.doi.org/10.1145/3290605.3300826>
- [44] Michael Nebeling, Janet Nebeling, Ao Yu, and Rob Rumble. 2018. ProtoAR: Rapid Physical-Digital Prototyping of Mobile Augmented Reality Applications. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, 353:1—353:12. DOI: <http://dx.doi.org/10.1145/3173574.3173927>
- [45] Michael Nebeling and Maximilian Speicher. 2018. The Trouble with Augmented Reality/Virtual Reality Authoring Tools. In *2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. 333–337. DOI: <http://dx.doi.org/10.1109/ISMAR-Adjunct.2018.00098>
- [46] NewGenApps. 2019. 6 VR and AR Statistics: Shaping the Future of Augmented Reality with Data. (2019). <https://www.newgenapps.com/blog/6-vr-and-ar-statistics-shaping-the-future-of-augmented-reality-with-data>
- [47] Benjamin Nuernberger, Eyal Ofek, Hrvoje Benko, and Andrew D Wilson. 2016. SnapToReality: Aligning Augmented Reality to the Real World. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 1233–1244. DOI: <http://dx.doi.org/10.1145/2858036.2858250>
- [48] Marc Rettig. 1994. Prototyping for tiny fingers. *Commun. ACM* 37, 4 (apr 1994), 21–27. DOI: <http://dx.doi.org/10.1145/175276.175288>
- [49] Nazmus Saquib, Rubaiat Habib Kazi, Li-Yi Wei, and Wilmot Li. 2019. Interactive Body-Driven Graphics for Augmented Video Performance. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, 622:1–622:12. DOI: <http://dx.doi.org/10.1145/3290605.3300852>
- [50] Hartmut Seichter, Julian Looser, and Mark Billinghurst. 2008. ComposAR: An Intuitive Tool for Authoring AR Applications. In *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR '08)*. IEEE Computer Society, Washington, DC, USA, 177–178. DOI: <http://dx.doi.org/10.1109/ISMAR.2008.4637354>
- [51] Bruce Tognazzini. 1994. The “Starfire” video prototype project. In *Proceedings of the SIGCHI conference on Human factors in computing systems celebrating interdependence - CHI '94*. ACM Press, New York, New York, USA, 99–105. DOI: <http://dx.doi.org/10.1145/191666.191712>
- [52] Peter Vistisen and Soren Bolvig Poulsen. 2017. Return of the Vision Video: Can corporate vision videos serve as setting for participation?. In *Nordic Design Research (NORDES)*. Nordes Digital Archive.
- [53] Pierre Wellner. 1993. Interacting with Paper on the DigitalDesk. *Commun. ACM* 36, 7 (jul 1993), 87–96. DOI: <http://dx.doi.org/10.1145/159544.159630>
- [54] Shihong Xia, Lin Gao, Yu-Kun Lai, Ming-Ze Yuan, and Jinxiang Chai. 2017. A Survey on Human Performance Capture and Animation. *Journal of Computer Science and Technology* 32, 3 (may 2017), 536–554. DOI: <http://dx.doi.org/10.1007/s11390-017-1742-y>
- [55] Robert Xiao, Chris Harrison, and Scott E Hudson. 2013. WorldKit: rapid and easy creation of ad-hoc interactive applications on everyday surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 879–888.