Montage: A Video Prototyping System to Reduce Re-Shooting and Increase Re-Usability

Germán Leiva, Michel Beaudouin-Lafon

Montage: A Video Prototyping System to Reduce Re-Shooting and Increase Re-Usability

Germán Leiva **Michel Beaudouin-Lafon**
Université Paris-Sud, CNRS, Inria, Université Paris-Saclay
91400 Orsay, France
{leiva, mbl}@lri.fr

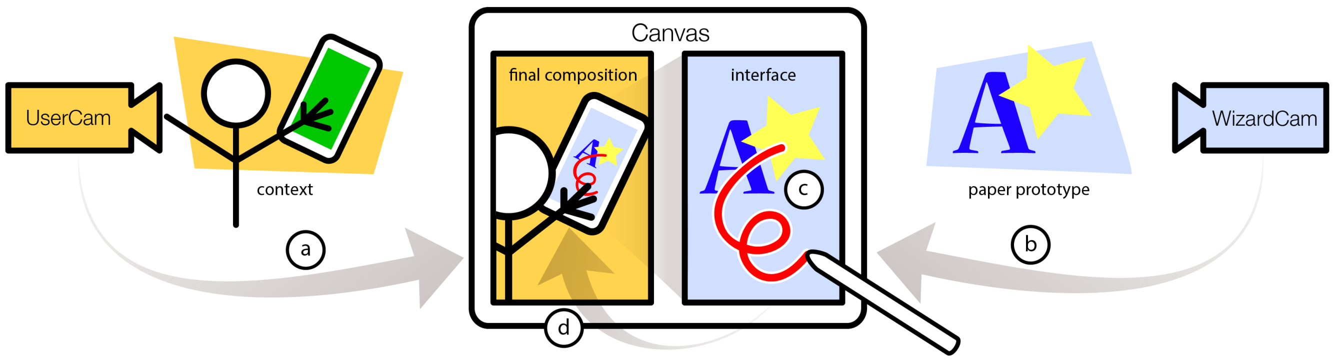

Figure 1: Chroma mode: the *UserCam* captures the *context* (a) and the *WizardCam* captures the *paper prototype* (b); Both live-stream video to the *Canvas*. Designers draw digital sketches over the streamed *paper prototype* to represent the *interface* (c). In the *Canvas*, the green screen is replaced with a perspective transformation of the *interface* to create the *final composition* (d).

ABSTRACT

Video prototypes help capture and communicate interaction with paper prototypes in the early stages of design. However, designers sometimes find it tedious to create stop-motion videos for continuous interactions and to re-shoot clips as the design evolves. We introduce Montage, a proof-of-concept implementation of a computer-assisted process for video prototyping. Montage lets designers progressively augment video prototypes with digital sketches, facilitating the creation, reuse and exploration of dynamic interactions. Montage uses chroma keying to decouple the prototyped interface from its context of use, letting designers reuse or change them independently. We describe how Montage enhances video prototyping by combining video with digital animated sketches, encourages the exploration of different contexts of use, and supports prototyping of different interaction styles.

CCS Concepts

•Human-centered computing → Systems and tools for interaction design; Interface design prototyping;

Author Keywords

Video prototyping; Paper prototyping; Wizard-of-Oz

Authors' version. Paper published at UIST '18, October 14–17, 2018, Berlin, Germany. DOI: https://doi.org/10.1145/3242587.3242613.

INTRODUCTION

Pen-and-paper is widely used when designing and communicating interactive systems, especially for quick prototyping [7]. Sketching on paper has well-known benefits [10]: it does not require technical skills, is inexpensive –in time and money– and, as a consequence, is easy to throw away. Paper excels in representing static visual properties and physical transformations such as moving paper elements [23]. However, paper makes it difficult or impossible to create dynamic transformations that continuously re-shape or modify the design elements, such as re-sizing or stretching elements or modifying colors and strokes in response to continuous user input.

In a paper prototyping session [42], a user interacts with the prototype, while one or more designers, or *wizards*, play the role of the computer. When the design changes or when exploring variants, instead of modifying the existing paper representations, they are thrown away and new ones are quickly created. The user can simulate the interaction over the paper prototype to communicate a rough idea, such as tapping with a finger to simulate a mouse click. The Wizard of Oz (WOz) technique [20] can create more realistic prototypes when the wizards conceal their actions. The WOz technique is not limited to paper, and can be used, e.g., with a video projector to create a more compelling setup.

Video prototyping [33, 34, 35] combines paper and video with the WOz technique to persist, communicate, and reflect about the interaction design. Videos can range from inexpensive recording of a traditional paper prototyping session [40] to a high-budget video prototype [43] requiring specialized equipment [5]. Video provides additional prototyping capabilities, such as jump cuts for simple appear/disappear effects or adding shots for contextualizing the user and the system within a story. However, using video together with paper hinders some of the benefits of using paper alone. Depending on the audience of the video, the wizard's trickery might need to be concealed, increasing the time and cost to produce a prototype. Introducing changes in the paper prototype creates inconsistencies with previously recorded scenes, leaving designers with three choices: sacrificing the consistency throughout the video, fixing the affected scenes in post-production editing, or re-shooting all the affected scenes.

Our goal is to provide better support for video prototyping in an integrated way to avoid inconsistencies [38]. How can we help designers persist their prototyping iterations consistently, with minimum post-production editing and re-shooting? We introduce Montage, a distributed mobile system supporting iterative video prototyping in the early stages of design. After reviewing related work, we describe Montage through a scenario that compares traditional video prototyping techniques with the enhanced approach using Montage. We then discuss prototyping opportunities with Montage for different interaction styles, including multi-modal interaction and augmented reality. Finally we describe the technical implementation of Montage, its current limitations, and future work.

RELATED WORK

In recent years many academic and commercial tools have emerged to support the prototyping of graphical user interfaces [41]. While pen-and-paper is one of "the most widely used prototyping medium" [13], some researchers argue that informal computer-based tools might better support the prototyping of interactive behaviors [4]. For example, SILK [26] lets designers sketch interfaces with a digital stylus to generate functional widgets, while Monet [29] expands this functionality to prototype continuous interactions by demonstration. Our goal is not to replace paper or impose an exclusive use of digital tools. Instead, Montage augments physical prototyping by extending the traditional paper-based techniques.

Other researchers have proposed tools that explicitly support the Wizard of Oz (WOz) technique. Some examples include WozARd for prototyping location-aware mobile augmented reality [1], SketchWizard for pen-based interfaces [15], and Suede for speech-based interfaces [25]. Apparition [27] helps designers prototype web-based systems in real time by crowdsourcing part of the wizard's trickery. Unlike these, Montage is not dedicated to a particular type of interface, making it a more generic tool for a variety of situations.

Furthermore, the WOz technique has several limitations, such as the wizard's stress and fatigue, the lack of reuse of the prototypes, and the delays and time lag between user actions and system response [39]. Montage supports live WOz but overcomes these shortcomings by using recorded and composable videos. Composition enables reuse while recording helps reduce timing and fatigue issues, e.g. wizards can pause and resume recording as needed.

We share similar goals with DART [32]: supporting early design stages and recording synchronized data. DART needs code for custom behaviors, but "interviewees consistently expressed a desire for a tool to support prototyping without coding" [19]. Unlike DART, Montage targets lower fidelity prototypes, does not require coding and accommodates other use cases besides augmented reality.

Montage is close to RPPT (Remote Paper Prototype Testing) [14] but serves a different purpose: RPPT is used to run live testing sessions with real users while Montage helps create reusable video prototypes with designers and explore alternatives. Like RPPT, Montage supports live streaming of paper prototypes. But Montage also persists the video prototype and lets designers modify the design after recording, e.g. by using time manipulation (rewind, pause, fast forward) or by composing different alternatives designs.

Commercial tools evolved from the graphic design tradition, starting from sketching tools but currently focusing on "pixel perfect" designs with graphic-authoring tools such as Adobe Illustrator or Photoshop. However, most of these tools do not target early-stage design as they focus on the final *look* [11] rather than the *feel*. The few tools that support nonstandard behaviors require visual [17] or textual [12] programming skills. Lee et al. [28] observes that much of the interactive behavior remains as textual descriptions due to the cost of creating dynamic prototypes, even for professionals.

Some tools extend traditional graphic authoring to support animations and effects, such as Flinto [18], but they ignore the role of user inputs and contexts of use. Only a handful of commercial tools support informal early-stage prototyping, e.g. by using paper-in-screen techniques [8]. For example, POP [30] lets designers create simple screen-flows by connecting pictures of paper prototypes through digitally defined hotspots. However, this only supports discrete actions, not dynamic interactions.

To solve these issues, many designers use presentation software, such as Keynote or Powerpoint, to mimic dynamic interactions [24]. While suitable for some use cases, e.g. WIMP and mobile apps, their pre-defined animations target effects and transitions among slides, covering a tiny subset of all the available interaction styles.

Professional designers also use video editing software, such as Adobe After Effects, to prototype the *look* of continuous interactions with high-fidelity videos. Luciani et al. [31] use animation-based sketching techniques with professional editing tools, such as Adobe Premiere. However, current approaches to video editing are complex and time-consuming, which conflicts with the goals of early-stage prototyping. VideoSketches [44] uses photos instead of videos to avoid the high cost and production issues of creating video scenarios. Dhillon et al. [16] have found no differences in the quality of feedback between a low-fidelity and a high-fidelity video. This supports the low-fidelity approach of Montage, based on freehand digital sketches and paper props. Montage directly supports an inexpensive animation-based sketching process, accessible to designers without video editing knowledge.

The benefits of low-fidelity video as a design tool have been investigated for a long time [36]. According to Greenberg et al., "design is putting things in context" [21]. Montage contextualizes the design by encouraging designers to be *user-actors* when demonstrating the prototype in a scenario [37].

In summary, current commercial tools create refined prototypes, more appropriate for mid/late stages of the design, while early-stage tools lack features to explore the details of continuous interaction. Montage fills this gap in the design space of prototyping tools: It enables the expression of highly dynamic interfaces in early low-fidelity video prototypes which can be recorded and modified without a need for post-production.

MONTAGE

Montage is composed of a central device –the *Canvas*– connected to two mobile devices –*UserCam* and *WizardCam*– with video streaming and recording capabilities. These devices, typically phones or tablets, are used as remote cameras. They stream, either in parallel or independently, the *context* of use where the user could interact with the prototype and the prototyped *user interface* itself. The *Canvas* lets designers organize and compose the video segments, and augment them with digital drawings that can be re-shaped and modified. Interaction designers can compose, draw and modify the prototype during rehearsal, during recording, or after filming. Montage focuses on low-budget video recording but provides designers with features currently only available in high-budget video prototyping, such as layering and tracking. Interaction designers can start with traditional paper prototyping and progressively move towards modifiable and re-usable digital representations without the need for professional video editing software.

Montage targets interactions that require continuous feedback, such as scaling a picture with a pinch or selecting objects with a lasso, which are often challenging to perform with traditional paper and video prototyping. We first illustrate the approaches and challenges of prototyping continuous feedback with traditional video prototyping, and then present an enhanced approach using Mirror, a mode of Montage that mixes streamed physical elements (such as a paper prototype) captured by a camera, with digital elements created remotely by a wizard. Finally, we present Montage Chroma to reduce re-shooting while exploring alternative designs.

Prototyping with traditional paper and video techniques

Imagine a group of designers prototyping an interaction technique with dynamic guides, similar to OctoPocus [6]. OctoPocus provides continuous feedback (inking) and feedforward (potential options) of the gestures as a user performs them[1]. The designers want to illustrate the use of this technique in the office, when interacting with the profile picture of a friend on a phone: when dwelling on the picture, OctoPocus should show three gestures for calling, messaging or finding directions to the friend. The designers print an image to use as the profile picture and attach it to the screen of a phone, used as a theatrical prop, to contextualize the prototyped interaction. The user-actor draws on the profile picture to mimic the continuous feedback of a gesture. She uses a black pen hidden as well as possible in his palm, while a wizard draws the feedforward, i.e. three colored curved lines.

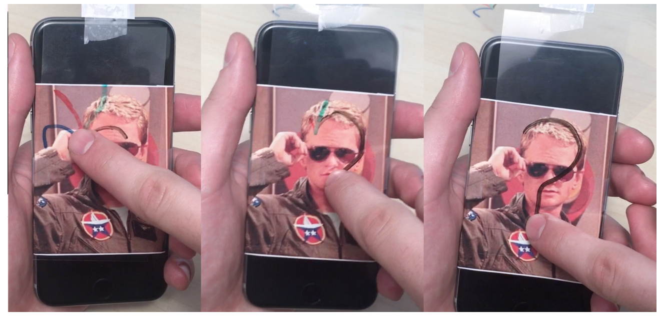

Figure 2. OctoPocus with traditional video prototyping. The designers create a rough stop-motion movie with only four stages of the interface, resulting in a poor representation of the dynamic interaction.

This approach to prototyping continuous feedback and feedforward has three main drawbacks:

- The hand and pen of the wizard appear in the video;
- The profile picture has drawings on it that might not be easy to erase, so in case of mistakes or changes, it requires a new picture or at least recording the whole video again; and
- Illustrating the use of the same technique in different contexts (on the profile picture of other friends, or in a completely different scenario) also requires re-shooting.

The designers take a different approach to avoid these problems. They create four sketches with transparent paper to represent the different stages of the OctoPocus interaction (Figure 2): They plan to reveal the feedforward and feedback progressively by overlaying the sketches on top of the profile picture, one sketch at a time. They use a mobile device on a tripod to record a rough stop-motion video of the interaction.

With this approach, the designers reduce the presence of the wizard in the video, as they place the sketches on top of the profile picture in-between the stop-motion takes. Because the sketches are drawn over transparent paper instead of the profile picture, the designers can reuse their prototype props to illustrate other contexts of use. Nevertheless, this approach also comes with limitations and drawbacks:

- While it is possible to reuse the sketches in other contexts, the whole interaction needs to be re-shot;
- A sequence of four sketches will poorly communicate the highly continuous nature of the interaction; and
- Making a stop-motion video shifts the designers' attention from experiencing and reflecting on their design to coordinating sequences of extremely brief video shots.

A third approach is to use video editing software instead of paper sketches to add a digital overlay with the user interface on top of a previously recorded shot of the user actions. This approach has the disadvantage of creating a disruptive context switch, from a design session that does not require specialized skills to a video editing session requiring trained editors. Also, with paper prototyping the user interface is partially hidden by the user's hands, so a simple digital overlay will not produce the same effect. Only an experienced video editor could sim-

[1] See https://vimeo.com/2116172

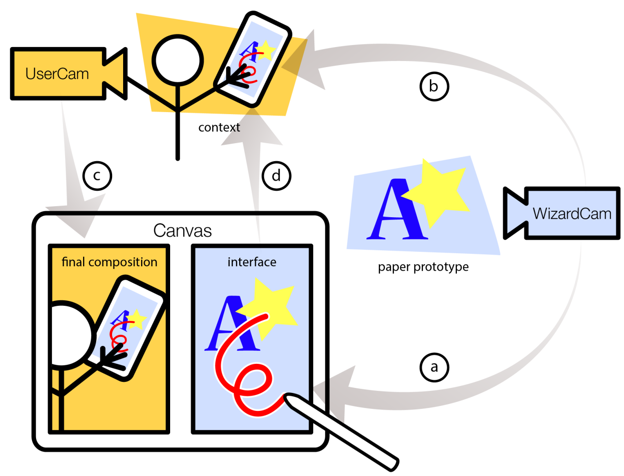

Figure 3. Montage Mirror: the *WizardCam* live streams to both, the *Canvas* (a) and to the prototyped device (b) in the *context*. The *UserCam* only streams to the *Canvas* (c). Finally, the *Canvas* sends the sketches to the prototyped device to complete the mirroring of the *interface*

ulate the fact that the interface is below the user's fingers by creating a mask of the user hands at several keyframes, e.g. by rotoscoping or using specialized software.

In summary, with current approaches to video prototyping, designers struggle to represent continuous feedback and to reuse prototype props and previously captured interactions. The tools that address these problems require video editing skills and extra effort in post-production, interrupting the flow of the design process. We want to better support video prototyping without disrupting the design process nor requiring specialized video editing skills.

Prototyping continuous feedback with Montage Mirror

Montage Mirror mixes physical elements captured by a *WizardCam* (Figure 3a and 3b), with digital sketches drawn remotely in the *Canvas* (Figure 3d). The user-actor's phone displays a video stream of the paper prototype combined with the digital sketches —the *interface*. As the user-actor interacts with the phone, the wizards provide live feedback by editing the digital sketches on the *Canvas* or by manipulating the paper prototype captured by the *WizardCam*.

For example, to prototype OctoPocus, the user-actor sees the profile picture on his phone, captured by the *WizardCam*. As she performs a gesture on the screen, she sees the feedback and feedforward sketched remotely by the wizard on the *Canvas*. In this way, the user-actor can experience the prototyped interaction without the hands of the wizard getting in the way. The *UserCam* captures the interaction over the *interface* and the *context* of use to create the final video prototype.

Designers can animate changes in position, size, rotation angle, color, and thickness of the digital sketches without the tedious coordination required by stop-motion videos. Digital sketches can be grouped to create compound objects that have a semantic meaning in the story. Moreover, thanks to the *WizardCam* stream, traditional paper prototyping techniques are still available if necessary: physical sketches and props added to the paper prototype are directly streamed to the user-actor's phone. For example, after the user-actor performs a gesture with OctoPocus, the wizard can add a sticky note with the text "Calling Barney" on top of the profile picture.

Montage Mirror augments video prototyping with *live* digital sketches. In the *Canvas*, designers use the stylus to draw sketches and perform simple actions, such as pressing a button. Designers can move, resize and rotate sketches with the standard pan, pinch and rotate gestures. Unlike stop-motion videos, digital sketches allow prototyping continuous feedback interactions that look fluid and allows designers to focus on the design process instead of coordinating complex wizard actions. For example, to prototype the drawing of a question mark and the inking feedback, the wizard draws at the same time that the user-actor is gesturing. The designer rewinds the recorded video to the point where she wants the dynamic guides to appear and draws them. After pressing play, she uses a slider of the sketch interface to make the stroke progressively disappear as the video plays (Figure 4).

Mirror mode supports the prototyping of dynamic interfaces and continuous feedback. Nevertheless, it still requires reshooting when exploring alternative designs or contexts, e.g., showing OctoPocus on something else than a mobile phone. Designers have to record the whole video again even for small changes, such as changing the color of the dynamic guides.

Montage Chroma: Reusing captured interactions

To help designers reuse previously captured interactions and reduce re-shooting, the Chroma mode takes advantage of a well-known video editing technique called *chroma key compositing*. With *chroma keying*, the subject is recorded in front of a solid background color, generally green or blue, and this background is replaced in post-production with the desired content. This technique is commonly used by weather presenters on television, to replace a green background with an animated map with weather information. In our example, the user drawing a question mark is recorded over a phone showing a green screen, which is later replaced with the prototype interface. Achieving a clean chroma keying requires special attention to proper lighting conditions and using the right shade of green. However, we are not concerned in achieving a perfect result during an early-stage low-fidelity prototype. We can also use a different color than green, as long as it is distinct enough from the rest of the scene, by selecting it in the video feed with Montage's color picker.

In order to replace only the portion of the screen that contains the *interface*, we display a green screen on the user-actor's phone. The *UserCam* records the final video prototype, but in Chroma mode Montage also tracks the four corners of the green screen and sends this data to the *Canvas*. Then, the *Canvas* performs a perspective transformation of the current frame of the *interface*, and replaces the green area with the transformed *interface*. Montage Chroma not only performs this composition in post-production, i.e. after recording, but also during recording, in the *final composition* live preview.

Designers simply need to rewind the recorded video to add new sketches or modify existing ones. They can draw or modify the

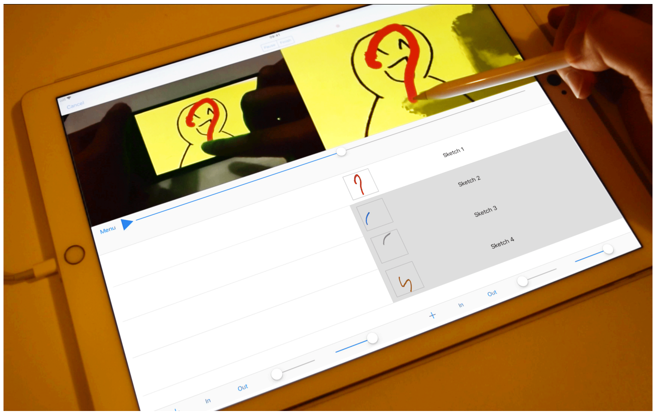

Figure 4. The *Canvas* sketching interface: The *final composition* (left) and the *interface* (right) show the "user overlay". Both sides have a list of sketches and animation controls at the bottom. The in/out buttons make the selected sketches appear/disappear. The sliders control the stroke-start, now at 0%, and the stroke-end, now at 100%.

sketches over the *interface* or over the *context* to annotate the story, e.g., with user reactions or speech bubbles. Designers do not need to re-shoot the *context* to make small changes to the *interface*, or vice versa. In the OctoPocus example, after recording a version of the prototype, designers can add new dynamic guides without re-shooting by simply drawing over the recorded video. The new changes are immediately available in the Canvas *final composition* live preview.

The setup of the system in Chroma mode (Figure 1) has just one difference from the setup in Mirror mode. The *UserCam* still captures the *context* but the phone now shows a green screen (Figure 1a). The *final composition* preview in the *Canvas* shows the chroma keyed result, positioning the *interface* correctly under the user's hands and following the phone's boundaries. When the designers modify the paper prototype or the digital sketches, i.e. add, remove, move, rotate, scale, or color change the sketches, Montage Chroma shows these modifications immediately in the *final composition* live preview, but not on the actual phone.

When possible, we recommend using Montage Mirror during rehearsal and Montage Chroma during recording. If the *interface* is to be placed on a screen-based device with streaming capabilities, using Montage Mirror lets the user-actor experience the prototype directly. During recording, using Montage Chroma allows to create composable video segments and lets the *interface* and the *context* to be changed independently. Montage Chroma provides a "user overlay" to assist wizards sketch in relation with user inputs. For example, when the user-actor places a finger over the green screen, the wizard can see a translucent image of this finger over the drawing area of the *Canvas*; this helps wizards better coordinate spatially and temporally the drawings with respect to the user inputs. Montage uses the inverse perspective transformation of the green screen to correct the placement of the overlay.

Exploring design alternatives and multiple contexts of use

With chroma keying, the recorded *interface* videos and *context* videos can be changed independently. This flexibility reduces the cost of exploring different design alternatives and multiple contexts of use. For example, once an *interface* video of the OctoPocus technique is prototyped, it can be embedded in multiple *contexts*: different videos can show the user-actor using OctoPocus in the metro, in a park or at a party without having to re-shoot the interaction technique. The other way around is also possible: Several alternative designs of the *interface* can be embedded in the same *context*, with different videos showing the original context of the actor-user in his office using OctoPocus on a social media profile, an email client or the camera app.

Besides recording with the *UserCam* in different places, i.e. in a park or an office, the *context* can be changed by using a different device. Montage Chroma works with any device that can display a solid color in a rectangular frame, such as watches, phones, tablets, laptops and even wall-size displays, enabling designers to explore multiple display alternatives.

Supporting multiple interaction styles

Montage Chroma is not limited to displays such as a phone's screen. Designers can video prototype over other rectangular surfaces, such as boxes, books and whiteboards with a solid color. We can use Montage to prototype gesture-based interactions over a green sticky note or a t-shirt stamp. This allows the exploration of stationary as well as mobile contexts, e.g., sitting in front of an interactive table or walking with a phone.

Montage's digital sketches can depict 2D widgets common in WIMP interaction such as buttons, sliders and menus. Toggling the visibility of sketches on or off (Figure 4) at precise moments in the video is ideal for prototyping interactive transitions of the interface states, e.g. idle/hover/press states of a button or the screen-flow of a mobile app. Static sketches can depict discrete feedback, e.g. adding an object after pressing a button, while animated sketches can depict continuous feedback, e.g. inking, dragging or resizing with a pinch.

Montage also supports prototyping interactions that involve indirect input devices. By using a green screen on a laptop's display we can still see the mouse cursor, facilitating the positioning of interface elements. For example, to prototype a marking menu, we create a simple paper prototype of the menu. When the user clicks, we pause recording and introduce the paper prototype under the *WizardCam*. We resume recording and when the user moves, the wizard adds digital sketches to illustrate the feedback. Finally, when the user selects an item, the wizard highlights the corresponding item and modifies the paper prototype to show the response of the system.

With Montage we can even prototype interactions that use the spatial relationship between the devices and the users, such as Proxemic Interaction [22]. For example, a user can walk closer or farther away from a device tracking her location, while the wizard manipulates the sketches to react to the user's position.

Prototyping multimodal interfaces, e.g., voice interaction and body movement, is possible with Montage. For example, to re-create the foundational Put-that-there interaction [9], both cameras record video and audio so that designers can enact the voice interactions that will be recorded. After the actor utters a voice command, the wizard pauses the recording, adds the necessary digital sketches and resumes recording. The video can then be modified, without re-shooting, to transition from paper to digital sketches or to add more details, such as the on-screen cursor. In more complex scenarios, the designers can sketch annotations on the *context* to indicate the user's voice commands or the system's audio responses, or to represent haptic feedback from the system, such as vibrations.

Video prototyping augmented reality systems is also easy with Montage. The designer attaches the *UserCam* to a headset, simulating the use of smart-glasses. With this setup, the user-actor has his hands free to interact with the overlaid images created by the wizard. In many cases, there is no need to use Chroma mode because the interface elements are overlaid over the camera feed. For example, to prototype a menu that follows the user's hand, the wizard only needs to sketch over the *context* and move the sketches to follow the hand's movements. The final video prototype will show the interaction from the point of view of the user-actor, i.e. the *UserCam*.

IMPLEMENTATION

Montage currently runs on iOS version 11.2. In our preferred setup, the *Canvas* runs on an iPad Pro 12.9 inches (2nd generation) with an Apple Pencil stylus. Generally, the *UserCam* runs on an iPhone 6S and the *WizardCam* on an iPad Mini 3. Other wireless devices are also suitable as cameras and mirrors. We tested Montage Mirror with an Apple Watch 1st gen. (42mm case) and a MacBook Pro (13-inch, 2015).

We use AVFoundation [2] to capture video, intercept frame buffers, and create movie files. The frame buffers are processed with Core Image [3] to clean the images before detecting rectangular areas, to perform perspective and correction transformations, and to execute the chroma keying.

We use a zero-configuration network where the *Canvas* acts as a server browsing for peers that automatically connect to the system. Each camera records its own high quality movie, currently 1280x720 pixels. However, the video stream is sent at a lower quality (480x360 pixels) to maintain an acceptable latency during rehearsal and recording (M=180ms, SD=60ms). The devices' clocks are synchronized to start, pause, and end the recording at the same time. Due to delays introduced by the wireless connection, we created a protocol to let the devices synchronize: When the designer presses *Record* on the *Canvas* the screen displays a *"3, 2, 1, go"* animation. This delay lets devices prepare for recording and synchronize their capture start time. We use the same mechanism when the designer resumes the recording after pausing.

In order to create a movie combining the dynamic sketches with the captured video, we save the designers' inputs during the manipulation of the digital sketches. We use this information to create keyframe animations at different points of the video playback. We added a synchronization layer on top of both to link these animated sketches with the underlining movie player. This new layer coordinates the animation playback with the movie file playback.

LIMITATIONS

We have observed that Chroma mode works best with flat surfaces, and rectangle tracking works poorly with flexible or shape-changing surfaces. Also, excessive user occlusion can prevent proper screen tracking. As a workaround, when the device is not moving, designers can lock the last tracked rectangle. Montage Chroma can also replace any solid-color area, regardless of its shape, with the *interface*. However, without position tracking, the perspective transformation of the interface is lost, resulting in a "naive" chroma keying.

One drawback of chroma keying is that the user-actor interacts with a green screen, not the final prototype. Using Mirror mode during rehearsal mitigates this problem. In Chroma mode, the user-actor should see the *Canvas* in order to monitor the state of the *interface* in relation with his inputs.

Finally, Montage only supports interaction styles that can be mimicked with video. Currently, we cannot illustrate complete immersive virtual worlds, e.g., VR applications or 3D games. However, Montage can still video prototype particular aspects of these examples, such as hand tracking.

CONCLUSION AND FUTURE WORK

Current approaches to video prototyping make it difficult to represent continuous feedback and reuse previously captured interactions. The tools that address these problems require video editing skills and extra effort in post-production, interrupting the flow of the design process. Therefore we need to better support video prototyping without disrupting the design process nor requiring specialized video editing skills.

We presented Montage, a distributed mobile system that supports video prototyping in the early stages of design. Our technical contribution is a novel tool integrating multiple live video streams, screen tracking, chroma keying, digital sketches and physical prototyping in one fully mobile video prototyping system. While these individual techniques are not new, Montage combines them in a novel way to support an enhanced video prototyping process: rehearsal with Mirror mode and recording with Chroma mode. Montage Mirror augments video prototypes with remote digital sketches, while Montage Chroma supports the reuse of previously recorded videos to explore alternative interfaces and contexts of use. We described a scenario that demonstrates the limitations of traditional paper and video techniques, and showed how Montage addresses them. Finally, we illustrated how Montage can prototype a variety of interaction styles, including touch-based (OctoPocus), WIMP (marking menu), AR (on-hand menu), multimodal (Put-that-there), and ubiquitous computing (Proxemic Interaction).

Future work involves evaluating Montage with professional interaction designers, improving screen tracking and exploring the reuse of video prototypes beyond early-stage design, e.g. by supporting the transition to high-fidelity prototypes.

ACKNOWLEDGMENTS

This work was partially supported by European Research Council (ERC) grants n° 321135 CREATIV: Creating Co-Adaptive Human-Computer Partnerships, and n° 695464 ONE: Unified Principles of Interaction. We thank Wendy Mackay and the ex)**situ** team for their help and support, and Carla Griggio for her multiple contributions to this work.

REFERENCES

1. Günter Alce, Klas Hermodsson, and Mattias Wallergård. 2013. WozARd: A Wizard of Oz Tool for Mobile AR. In *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services - MobileHCI '13*. ACM Press, New York, New York, USA, 600. DOI: http://dx.doi.org/10.1145/2493190.2494424
2. Apple Inc. 2009. Apple Developer, AV Foundation. (2009). Retrieved July 12, 2018 from https://developer.apple.com/av-foundation/.
3. Apple Inc. 2011. Apple Developer Documentation, Core Image. (2011). Retrieved July 12, 2018 from https://developer.apple.com/documentation/coreimage.
4. Brian P. Bailey and Joseph A. Konstan. 2003. Are informal tools better?: comparing DEMAIS, pencil and paper, and authorware for early multimedia design. In *Proceedings of the conference on Human factors in computing systems - CHI '03*. ACM Press, New York, New York, USA, 313. DOI: http://dx.doi.org/10.1145/642611.642666
5. 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
6. Olivier Bau and Wendy E. Mackay. 2008. OctoPocus: A Dynamic Guide for Learning Gesture-based Command Sets. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology (UIST '08)*. ACM, New York, NY, USA, 37–46. DOI: http://dx.doi.org/10.1145/1449715.1449724
7. 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
8. Davide Bolchini, Diego Pulido, and Anthony Faiola. 2009. "Paper in Screen" Prototyping: An Agile Technique to Anticipate the Mobile Experience. *interactions* 16, 4 (jul 2009), 29. DOI: http://dx.doi.org/10.1145/1551986.1551992
9. Richard A Bolt. 1980. *"Put-that-there": Voice and gesture at the graphics interface*. Vol. 14. ACM.
10. 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
11. Bohemian B.V. 2009. Sketch. (2009). Retrieved July 12, 2018 from https://www.sketchapp.com.
12. Framer BV. 2015. Framer. (2015). Retrieved July 12, 2018 from https://framer.com/.
13. 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
14. 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
15. Richard C. Davis, T. Scott Saponas, Michael Shilman, and James A. Landay. 2007. SketchWizard: Wizard of Oz Prototyping of Pen-Based User Interfaces. In *Proceedings of the 20th annual ACM symposium on User interface software and technology - UIST '07*. ACM Press, New York, New York, USA, 119. DOI: http://dx.doi.org/10.1145/1294211.1294233
16. B. Dhillon, P. Banach, R. Kocielnik, J. P. Emparanza, I. Politis, A. Pączewska, and P. Markopoulos. 2011. Visual Fidelity of Video Prototypes and User Feedback: A Case Study. In *Proceedings of the 25th BCS Conference on Human-Computer Interaction (BCS-HCI '11)*. British Computer Society, Swinton, UK, UK, 139–144. http://dl.acm.org/citation.cfm?id=2305316.2305342
17. Facebook. 2013. Origami. (2013). Retrieved July 12, 2018 from https://origami.design/.
18. Flinto. 2010. Flinto. (2010). Retrieved July 12, 2018 from https://www.flinto.com/.
19. 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
20. 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
21. Saul Greenberg, Sheelagh Carpendale, Nicolai Marquardt, and Bill Buxton. 2012. The Narrative Storyboard Telling a story about use and context over time. *interactions* 19, 1 (jan 2012), 64. DOI: http://dx.doi.org/10.1145/2065327.2065340
22. Saul Greenberg, Nicolai Marquardt, Till Ballendat, Rob Diaz-Marino, and Miaosen Wang. 2011. Proxemic Interactions: The New Ubicomp? *interactions* 18, 1 (jan 2011), 42–50. DOI: http://dx.doi.org/10.1145/1897239.1897250
23. Saul Greenberg, Carpendale Sheelagh, Marquardt Nicolai, and Buxton Bill. 2012. *Sketching User Experiences: The Workbook*. Morgan Kaufmann. 272 pages. DOI:http://dx.doi.org/10.1016/C2009-0-61147-8

24. Khella Productions Inc. 2013. Keynotopia. (2013). Retrieved July 12, 2018 from https://keynotopia.com/.

25. Scott R. Klemmer, Anoop K. Sinha, Jack Chen, James A. Landay, Nadeem Aboobaker, and Annie Wang. 2000. Suede: A Wizard of Oz Prototyping Tool for Speech User Interfaces. In *Proceedings of the 13th annual ACM symposium on User interface software and technology - UIST '00*. ACM Press, New York, New York, USA, 1–10. DOI:http://dx.doi.org/10.1145/354401.354406

26. 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

27. Walter S Lasecki, Juho Kim, Nicholas Rafter, Onkur Sen, Jeffrey P Bigham, and Michael S Bernstein. 2015. Apparition: Crowdsourced User Interfaces That Come To Life As You Sketch Them. (2015). DOI: http://dx.doi.org/10.1145/2702123.2702565

28. Sang Won Lee, Yujin Zhang, Isabelle Wong, Yiwei Yang, Stephanie D. O'Keefe, and Walter S. Lasecki. 2017. SketchExpress: Remixing Animations for More Effective Crowd-Powered Prototyping of Interactive Interfaces. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology - UIST '17*. ACM Press, New York, New York, USA, 817–828. DOI: http://dx.doi.org/10.1145/3126594.3126595

29. Yang Li and James A. Landay. 2005. Informal prototyping of continuous graphical interactions by demonstration. In *Proceedings of the 18th annual ACM symposium on User interface software and technology - UIST '05*. ACM Press, New York, New York, USA, 221. DOI:http://dx.doi.org/10.1145/1095034.1095071

30. MarvelApp Prototyping Ltd. 2012. POP. (2012). Retrieved July 12, 2018 from https://marvelapp.com/pop/.

31. Danwei Tran Luciani and Peter Vistisen. 2017. Empowering Non-Designers Through Animation-based Sketching. In *7th Nordic Design Research ConferenceNordic Design Research Conference*, Vol. 7. http://www.nordes.org/nordes2017

32. 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

33. Wendy E Mackay. 1988. *Video Prototyping: a technique for developing hypermedia systems*. Vol. 5. ACM/SIGCHI.

34. Wendy E. Mackay. 2002. *Using Video to Support Interaction Design*. INRIA Multimedia Services. https://www.lri.fr/~mackay/VideoForDesign/

35. Wendy E. Mackay and Anne-Laure Fayard. 1999. Video brainstorming and prototyping: techniques for participatory design. *CHI'99 extended abstracts on Human factors in Computing Systems* May (1999), 118–119. DOI:http://dx.doi.org/10.1145/632716.632790

36. Wendy E. Mackay, R. Guindon, M. M. Mantel, Lucy Suchman, and D. G. Tatar. 1988. Video: Data for Studying Human-Computer Interaction. *Chi'88* (1988), 133–137. DOI:http://dx.doi.org/10.1145/57167.57189

37. Wendy E. Mackay, Anne V. Ratzer, and Paul Janecek. 2000. Video artifacts for design: bridging the Gap between abstraction and detail. In *DIS '00*. ACM, New York, New York, USA, 72–82. DOI: http://dx.doi.org/10.1145/347642.347666

38. Nolwenn Maudet, Germán Leiva, Michel Beaudouin-Lafon, and Wendy E. Mackay. 2017. Design Breakdowns: Designer-Developer Gaps in Representing and Interpreting Interactive Systems. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing - CSCW '17*. ACM Press, New York, New York, USA, 630–641. DOI: http://dx.doi.org/10.1145/2998181.2998190

39. John Sören Pettersson and Malin Wik. 2014. *Perspectives on Ozlab in the cloud: A literature review of tools supporting Wizard-of-Oz experimentation, including an historical overview of 1971-2013 and notes on methodological issues and supporting generic tools*. Technical Report. Karlstad University, Karlstad Business School. 109 pages. http://urn.kb.se/resolve?urn=urn%3Anbn%3Ase%3Akau%3Adiva-33617

40. Marc Rettig. 1994. Prototyping for tiny fingers. *Commun. ACM* 37, 4 (apr 1994), 21–27. DOI: http://dx.doi.org/10.1145/175276.175288

41. Thiago Rocha Silva, Jean-Luc Hak, Marco Winckler, and Olivier Nicolas. 2017. A Comparative Study of Milestones for Featuring GUI Prototyping Tools. *Journal of Software Engineering and Applications* 10, 06 (jun 2017), 564–589. DOI: http://dx.doi.org/10.4236/jsea.2017.106031

42. Carolyn Snyder. 2004. *Paper prototyping: The fast and easy way to design and refine user interfaces*. Morgan Kaufmann. 408 pages. DOI: http://dx.doi.org/10.1016/B978-1-55860-870-2.X5023-2

43. 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

44. John Zimmerman. 2005. Video Sketches: Exploring Pervasive Computing Interaction Designs. *IEEE Pervasive Computing* 4, 4 (Oct. 2005), 91–94. DOI: http://dx.doi.org/10.1109/MPRV.2005.91