360Anywhere: Mobile Ad-hoc Collaboration in Any Environment using 360 Video and Augmented Reality

MAXIMILIAN SPEICHER, University of Michigan, USA
JINGCHEN CAO, University of Michigan, USA and Fudan University, China
AO YU, University of Michigan, USA and Tsinghua University, China
HAIHUA ZHANG, University of Michigan, USA and Tsinghua University, China
MICHAEL NEBELING, University of Michigan, USA

360-degree video is increasingly used to create immersive user experiences; however, it is typically limited to a single user and not interactive. Recent studies have explored the potential of 360 video to support multi-user collaboration in remote settings. These studies identified several challenges with respect to 360 live streams, such as the lack of gaze awareness, out-of-sync views, and missed gestures. To address these challenges, we created *360Anywhere*, a framework for 360 video–based multi-user collaboration that, in addition to allowing collaborators to view and annotate a 360 live stream, also supports projection of annotations in the 360 stream back into the real-world environment in real-time. This enables a range of collaborative augmented reality applications not supported with existing tools. We present the 360Anywhere framework and tools that allow users to generate applications tailored to specific collaboration and augmentation needs with support for remote collaboration. In a series of exploratory design sessions with users, we assess 360Anywhere's power and flexibility for three mobile ad-hoc scenarios. Using 360Anywhere, participants were able to set up and use fairly complex remote collaboration systems involving projective augmented reality in less than 10 minutes.

CCS Concepts: • **Human-centered computing → Mixed / augmented reality**; *Computer supported cooperative work*; *Mobile devices*;

Additional Key Words and Phrases: 360 video; augmented reality; mobile devices; remote assistance; remote collaboration

ACM Reference Format:
Maximilian Speicher, Jingchen Cao, Ao Yu, Haihua Zhang, and Michael Nebeling. 2018. 360Anywhere: Mobile Ad-hoc Collaboration in Any Environment using 360 Video and Augmented Reality. *Proc. ACM Hum.-Comput. Interact.* 2, EICS, Article 9 (June 2018), 20 pages. https://doi.org/10.1145/3229091

1 INTRODUCTION

360-degree video introduces many benefits for collaboration compared to regular video chat, especially for remote collaborators [23]. A remote collaborator has total autonomy over the video scene while a local collaborator is no longer responsible for performing tedious camerawork. Widely available and inexpensive cameras like RICOH's *Theta* make the advantages of 360-degree video

Authors' addresses: Maximilian Speicher, University of Michigan, Ann Arbor, MI, 48109, USA, mspeiche@umich.edu; Jingchen Cao, University of Michigan, Ann Arbor, USA, Fudan University, Shanghai, China, jccao14@fudan.edu.cn; Ao Yu, University of Michigan, Ann Arbor, USA, Tsinghua University, Beijing, China, ya14@mails.tsinghua.edu.cn; Haihua Zhang, University of Michigan, Ann Arbor, USA, Tsinghua University, Beijing, China, haihuaz@yeah.net; Michael Nebeling, University of Michigan, Ann Arbor, USA, nebeling@umich.edu.

2573-0142/2018/6-ART9 $15.00
https://doi.org/10.1145/3229091

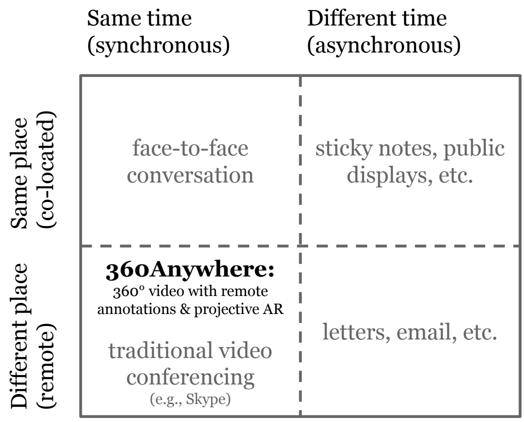

Fig. 1. The collaboration scenarios targeted by 360Anywhere lie in the "same time, different place" quadrant of the Time/Space Matrix, i.e., synchronous, remote collaboration.

accessible to a wide range of consumers and enable live streaming in mobile settings. However, existing work on potential benefits of using 360-degree video (e.g., [14, 22, 23]) for collaboration has previously only considered regular interaction via live video, rather than placing annotations—in the form of drawings, text, or images—directly in the 360-degree video stream, which is possible with 360Anywhere. On the other hand, research on remote collaboration with annotations (e.g., [3, 5, 12]) so far focused on plain video use cases, not taking advantage of the benefits of 360-degree video. Tang *et al.* [23] point out that 360-degree video introduces a potentially problematic asymmetry between the local and remote collaborators and have identified three related, unresolved challenges: *(C1: Gaze)* because of individual perspectives, it is not clear which portion of the 360 video collaborators are seeing; *(C2: Out-of-sync)* unlike in 2D video, in 360-degree video collaborators do not necessarily share the same view; and *(C3: Gestures)* gestures performed by one collaborator may be missed by others not sharing the same view.

To address these issues, we present *360Anywhere* (Fig. 2), a 360-degree video-based mobile ad-hoc collaboration framework with support for projective augmented reality (AR). Based on 360-degree input and projective AR, our framework provides a range of configurable components, such as session support, different kinds of annotations, and gaze awareness, from which the collaborators can choose, and enables distribution to a variety of output devices. For this, local collaborators, say in a meeting room, first set up a 360-degree camera connected to a computer and specify its live stream as the input for 360Anywhere in the configuration step. Remote collaborators can then access the live stream showing the meeting room simply via their mobile phones by opening a website provided by the system. With our framework, however, they are no longer restricted to just consuming a 360-degree video, but can now also provide input directly through the stream—in the form of drawings, text, and images—which is then synchronized between all collaborators. In particular, when one or more projections are available, these can be defined within the 360-degree live stream, so that remote collaborators can draw and place text and images directly in the room through projective AR. For instance, local participants in the meeting room set up a projection on a whiteboard. In the 360-degree live stream, in which the projection is now also visible, they click its four corners for calibration. Next, they place a number of designs on the whiteboard and ask remote collaborators for their preference. The remote collaborators can circle their favorites directly in the stream. These annotations appear on the whiteboard in real-time through the projection.

In an exploratory study structured into three design jams with interaction design Master's students (N=10), we evaluated the feasibility and effectiveness of our 360Anywhere framework for a range of mobile, ad-hoc scenarios. These scenarios are located in the "same time, different place" quadrant of the Time/Space Matrix (Fig. 1). Participants were divided into three groups and provided with different devices. All of the groups were introduced to three scenarios and asked to reason about possible configurations for 360Anywhere while generating remote collaboration applications for those scenarios. We also elicited additional scenarios in which they deemed different features of the framework particularly useful compared to existing solutions. Results indicate that the above challenges can be solved using our framework and that users are able to rapidly design and deploy 360-degree collaboration setups.

With 360Anywhere, this paper makes three primary contributions:

(1) To support synchronous, remote collaboration in any environment, we provide a framework that extends 360-degree video with a set of multi-user awareness features and annotation tools (Sec. 3.1), and support for mapping 360-degree video annotations back to projections in the real-world environment (Fig. 4).

(2) To support a variety of collaboration scenarios and user needs (e.g., 1-to-1, 1-to-many, many-to-many), we build on a component-based architecture (Fig. 3) and identify and implement the minimal set of required framework components (Sec. 3.2).

(3) To support a range of configurations and mobile ad-hoc collaboration scenarios, we explore the power and flexibility of our framework through a user study with 10 participants and three scenarios in a dynamic setup. Our evaluation shows that 360Anywhere:

a) can mitigate existing challenges with 360-based collaboration (e.g., gaze awareness);
b) is feasible for use with mobile devices in ad-hoc scenarios; and
c) can be configured for various collaboration scenarios.

The technical contribution of our work is the 360Anywhere system. In line with Olsen [17], we show that 360Anywhere addresses unresolved challenges, reduces solution viscosity, empowers new design participants, and leverages power in combination. While out of scope here, subsequent work may explore the impact of systems like 360Anywhere on users' collaborative behavior.

2 RELATED WORK

The research presented in this paper combines work from three so far independent research streams—*projective augmented reality*, *360-degree video*, and *remote collaboration*. We present a representative sample of related research and point out the differences to our proposed solution.

2.1 Projective Augmented Reality

Our work on 360Anywhere adds to existing research on smart rooms and multi-display environments, e.g., [2, 6, 8, 18, 20]. Most closely related is the set of research projects carried out by Wilson and Benko on projective augmented reality. With *LightSpace*, Wilson *et al.* [25] describe a room equipped with multiple depth sensors and projectors that allow physical surfaces to be turned into interactive displays. In [9], Jones *et al.* introduce *IllumiRoom*, which extends the space around a television by augmentation in order to complement gaming and other entertainment experiences. The *RoomAlive* project [10] builds on a set-up with multiple projector–depth camera units to map a whole room and turn it into an immersive entertainment experience. *MeetAlive* [4] is a room-scale omni-directional display system involving multiple depth cameras and projectors that lets users share content from their computers at any place in the room that has been equipped. Anyone participating in the meeting can then interact with all of the shared content through a perspective-corrected mouse cursor.

While these systems are impressive in their ability to transform physical rooms into interactive multi-display environments, they all have in common that they require a good amount of calibration and instrumentation of user and/or environment, and are therefore not designed for mobile and ad-hoc collaborative settings. With our work on 360Anywhere, we were prepared to trade complexity for flexibility and aimed at a lower fidelity solution. We want to enable end-users with an easy and quick set-up with minimal and affordable hardware requirements, namely a 360-degree camera and a mobile projector, and in this way, rather than instrumenting the room itself with interactive capabilities, we provide a virtual smart room in the 360-degree stream.

2.2 360-degree Video

As previously mentioned, 360-degree video is receiving increased attention in research where it has so far been studied primarily as a medium to consume immersive media content. For example, in recent work by Lin *et al.* [14], two techniques are described for directing a viewer's focus to certain points of interest in a 360-degree video: autopilot and visual guidance.

On the other hand, Tang *et al.* [22] started to investigate how multiple users can effectively view 360-degree videos together in a guided tour scenario. Another project by Tang *et al.* [23] that we already mentioned in the introduction explored how pairs of users collaboratively experience environments as they are being live streamed from one participant to another via a 360-degree camera. Tang *et al.* identified several challenges, the most important ones of which include *gaze awareness*, *out-of-sync views*, and *missed gestures*.

In terms of gaze awareness, there is research that looked at possibilities to achieve bidirectional gaze awareness, e.g., based on regular webcams [13]. In ParaFrustum [21], Sukan *et al.* explored visualization techniques designed to guide users to a specific set of viewing positions and orientations in virtual or augmented reality interfaces.

While these works investigated 360-degree video and, to some degree, explored the benefits and trade-offs for collaborative scenarios, there is no prior system that addresses the issues in a comprehensive manner. Motivated by this lack of support, with 360Anywhere, we wanted to explore how to best enhance 360-degree live streams for remote collaboration in augmented reality. Our framework incorporates components for remote collaboration that the above systems lack.

2.3 Remote Collaboration

In research on remote collaboration, one of the most important requirements is support for remote annotations. A number of recent papers [3, 5, 12] investigated remote annotations in mobile scenarios, with a particular focus on scene exploration and how to enable the remote collaborator to provide stable annotations with shaky camera feeds. While these papers provide valuable input for parts of our implementation when it comes to freezing the live feed upon input, they do not consider support for 360-degree video or projective AR and the additional issues this introduces.

There is existing work on remote collaboration in augmented reality that is motivational to 360Anywhere. For example, *Project Chalk* by Vuforia [24] provides tools for remote assistance in ad-hoc scenarios based on a mobile app, so that users can assist remote collaborators with their regular, everyday smartphones.

Skype on HoloLens [15] takes a similar approach, where a remote collaborator with, e.g., a tablet PC can annotate the environment of the HoloLens user by drawing on walls and tables, among other things. This solution provides AR capabilities similar to 360Anywhere, but is limited to 1-on-1 calls. Also, it introduces the exact problems with standard, non-360-degree video collaboration described by [23] (i.e., remote collaborators have little autonomy and the HoloLens user is responsible for framing objects, effective overviews, steady camerawork etc.).

To give a last example, *Sketchboard.io* is designed to be a collaborative sketching tool. However, compared to 360Anywhere, it does not feature video collaboration or AR capabilities for annotating an environment.

Most existing annotation tools have in common that they build on 2D annotations drawn on a virtual or physical screen. Sodhi *et al.* [19] investigated possibilities to provide 3D and spatial input for remote collaboration. However, this solution has additional hardware requirements in the form of a depth camera that we considered to be too limiting from a practical standpoint. In 360Anywhere, local collaborators can draw directly on physical whiteboards and remote collaborators can annotate a 360 live stream while the system deals with handling the live stream annotations on spherical video and perspective projection back into the physical world.

Other research has, e.g., looked at recreating parts of an environment from a wearable camera for a remote observer [1] and collaborative AR for mathematics and geometry education [11]. Yet, such solutions have specialized use cases and therefore limited flexibility and require sophisticated hardware set-ups. Also, they do not leverage the advantages of 360-degree video for their existing telepresence or AR solutions.

Informed by this existing research, we present a first framework that integrates *360-degree video* with *projective augmented reality*. Overcoming many of the shortcomings of existing systems with *remote annotations* of 360 live streams and tool support to address the challenges of *gaze awareness*, *out-of-sync views*, and *missed gestures*, 360Anywhere enables a variety of multi-user scenarios with local and remote collaborators in an ad-hoc manner.

3 360ANYWHERE

360Anywhere is a configurable framework to enable remote collaboration in augmented reality, based on 360-degree video input. It provides end-users with means for the flexible and quick set-up of a corresponding system, particularly in mobile and ad-hoc settings. The only technical requirement is a 360-degree camera that is capable of live streaming into a web application. Our framework consists of three parts: *a)* a **Configuration** for defining and setting up the remote collaboration system; *b)* a **Collaborator UI** for direct interaction through the 360-degree live stream; and *c)* a **Projection UI** for digital annotation of the real-world environment.

3.1 Configuration

The user's entry point for defining a set-up for a remote collaboration system is 360Anywhere's Configuration (cf. Figure 2, ①). Configuration happens on the device that will be used for broadcasting, i.e., the device has to be connected to the 360-degree camera. Through a simple UI the user has to provide information that is necessary to configure the system and tailor it to the collaborators' needs. First, the user has to select the correct camera, so that the 360-degree live stream can be broadcasted. Second, they state whether projections are present in the environment to be broadcasted, and if so, how many. Projections are necessary if augmenting the real-world environment with digital annotations is intended. Third, they decide on the optional components that shall be included in the system. The available optional components are: *Audio/Video Chat*, *Gaze Awareness*, *Follow Me*, *Back in Time*, and *Session Support*.

In case the number of projections specified by the user is greater than zero, two additional components are automatically included in the system, i.e., *Calibration* and *Transformation*. The first enables the definition of projections within the 360-degree stream; the second is responsible for correctly displaying annotations from the stream in projections in the real-world environment. The *Annotation* components (drawings, text, and images) are always included in the system.

Once the user has provided all of the necessary input for setting up the system, they are provided with *a)* a customized URL to be used by the local collaborators; *b)* a customized URL for

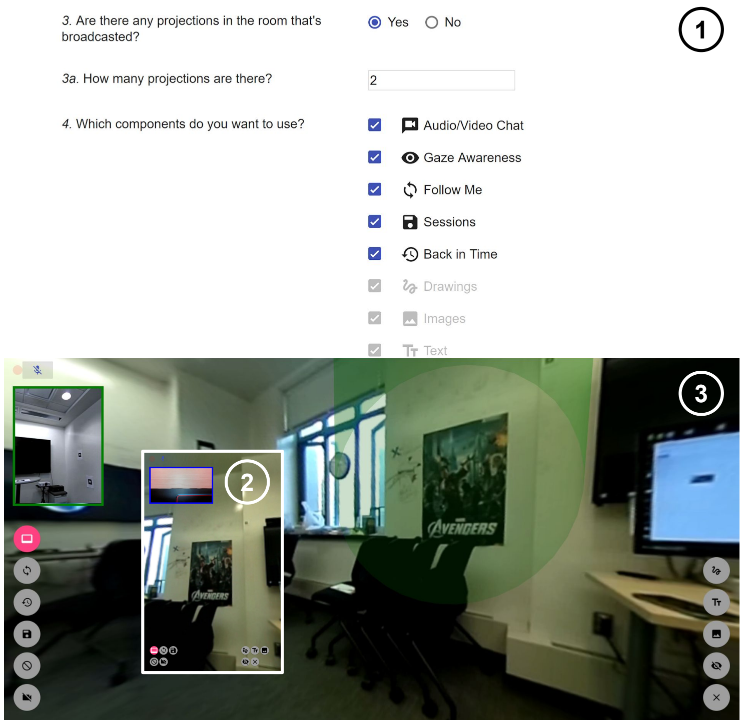

Fig. 2. (1) 360Anywhere's *Configuration* with activated framework components; (2) the *Collaborator UI* as seen by a remote participant; (3) the *Collaborator UI* with *Gaze Awareness* as seen by a local participant.

remote collaborators; and *c)* individual URLs for all of the projections present in the broadcasted environment (if any). The projections are numbered to make them easily identifiable (#1 for the 1st projection, #2 for the 2nd, and so on).

3.2 Collaborator UI

Once 360Anywhere has been set up, the Collaborator UI provides the live stream broadcasted by the 360-degree camera along with a UI for interacting with the components contained in the system, as per the configuration (cf. Figure 2, ②③). Each remote collaborator who accesses the UI is assigned a different, unique color. In the following we describe the functionality of each component that a collaborator can directly interact with through the Collaborator UI.

In the introduction, we have already described challenges with 360-degree video collaboration identified by [23]:

(C1: Gaze) It is not clear which portion of the 360 video collaborators are seeing.
(C2: Out-of-sync) In 360-degree video collaborators do not necessarily share the same view.
(C3: Gestures) Gestures performed by one collaborator may be missed by others not sharing the same view.

Besides these, there are additional challenges that arise from the use of video streams and remote annotations, i.e.,

(C4: Attention) collaborators that do not pay attention, and

(C5: Persistence) persisting digital annotations that were added to the real-world environment at their exact positions, e.g., drawings on a whiteboard (cf. Fig. 5, ④).

The components included in the 360Anywhere framework were designed to provide solutions for all of these challenges.

Gaze Awareness. Displays colored cones that indicate where each collaborator is looking in the 360-degree stream (cf. Figure 2, ②). The color of a cone corresponds to the unique color of the collaborator. (Intended to address *C1: Gaze* and *C2: Out-of-sync.*)

Follow Me. Enables one collaborator to gain control of everyone's 360 feed, so that the direction of viewing is synchronized across all collaborators. (Intended to address *C1: Gaze* and *C2: Out-of-sync.*)

Audio/Video Chat. Provides a separate Skype-like channel (cf. Figure 2, ②③) for additional audio-visual communication, e.g., hand gestures. (Intended to address *C1: Gaze*, *C2: Out-of-sync*, and *C3: Gestures.*)

Back in Time. Enables the remote collaborator to rewind the live stream by 10 seconds, e.g., in case they missed something important. (Intended to address *C4: Attention.*)

Session Support. Provides functionality for saving all digital artifacts in a session at their exact positions, as well as a snapshot of the current 360-degree stream (including all annotations and projections). Sessions can be loaded at a later point in time even if the system was restarted. (Intended to address *C5: Persistence.*)

Annotations. Provides functionality for drawing, placing images, and writing text directly into the 360-degree stream. (Intended to address *C1: Gaze* and *C2: Out-of-sync.*)

Calibration. Enables the user to define one or more projections in the 360-degree live feed. This happens by choosing which projection to define (in terms of its unique number), then clicking the four corners of the projection in the stream (displayed by the Projection UI). All annotations that lie within the defined borders are then also displayed in the corresponding Projection UI in the real-world environment. (Intended to address *C1: Gaze*, *C2: Out-of-sync*, and *C5: Persistence.*)

All annotations and projections are automatically synchronized across all local and remote collaborators. That is, everybody can see everybody else's annotations in real time. Moreover, when one collaborator has defined a projection in the 360-degree stream, that calibration is propagated to the other collaborators to eliminate the need for re-calibration.

3.3 Projection UI

360Anywhere's Projection UI comes into play when one or more projections are present in the environment being broadcasted (cf. Fig. 5, ④⑦). All projectors have to display one of the unique Projection UIs given by the Configuration in full-screen mode. During calibration, the Projection UI highlights, one after the other, its four corners, which the user has to click in the 360-degree live feed. Then, when the projections have been calibrated in the Collaborator UI, all annotations placed or drawn within the boundaries of a projection in the 360-degree stream are automatically propagated to the corresponding Projection UI in real time. This enables collaborators to annotate a real-world environment with digital artifacts that can also be persisted if the session component is active.

4 ARCHITECTURE AND IMPLEMENTATION

Our aim is to provide a remote AR collaboration framework that works in a wide range of scenarios, with a particular focus on mobile, ad-hoc use cases. This poses certain requirements, i.e., the intended solution must be *flexible*, *lightweight*, and *device-agnostic.* Therefore, 360Anywhere follows

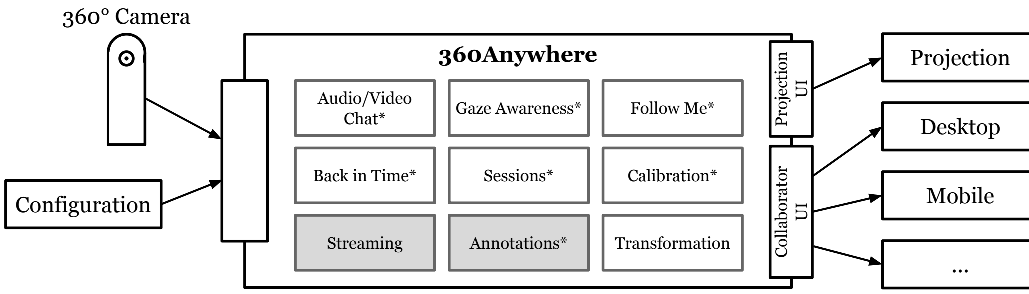

Fig. 3. The component-based architecture of our framework. It takes a 360-degree stream and a user-provided configuration as input (left) and outputs a corresponding remote collaboration system that can be used with a variety of devices (right). The two components highlighted in gray are always contained in the system; the remaining components can be activated/deactivated by the user to tailor the result to their needs. Through the Collaborator UI, the user can directly interact with the components marked with an asterisk.

a component-based approach and has been designed to prevent overhead and provide remote collaboration systems that are tailored to the collaborators' needs.

The inputs required by the framework are a 360-degree video stream and a configuration defined by the user (i.e., the optional components to be included, the number of projections, etc.). The output of the framework is a ready-to-use remote collaboration system that can then be deployed on the collaborators' devices as well as any computers providing projections in the broadcasted environment—simply by opening the provided URLs in a web browser. Two components—Streaming and Annotations—are always part of such a system since they provide the underlying functionality of 360Anywhere (Figure 3, highlighted in gray). The remaining components build on that underlying functionality, but are not crucial to the system and can therefore be activated or deactivated based on the collaborators' needs. For instance, if the collaborators have to work with low-end smartphones, Audio/Video Chat would have a negative impact on performance if activated. In other scenarios, collaborators might not need Gaze Awareness or Follow Me, which makes for a more lightweight experience and a decluttered UI if deactivated.

360Anywhere is based on a client-server architecture. The user interacts with the framework through the aforementioned web interfaces—Configuration, Collaborator UI, and Projection UI. These are provided by a server that is also responsible for communication between the different parts of the system, which has been realized using *Socket.IO* and *WebRTC*. On the server side, the framework has been implemented using *Node.js* while on the client side, we build on state-of-the-art web technologies, including *A-Frame* and *three.js*. All of 360Anywhere's optional components are independent JavaScript modules that reside on the client side. Contrarily, the two underlying components are distributed JavaScript modules spread across the server and the Collaborator UI—they are the "glue" of the framework. By building on web technologies and providing responsive UIs, we provide a device-agnostic framework, a requirement for supporting mobile, ad-hoc scenarios.

In the following we describe the defining parts of our framework in more detail.

4.1 Streaming

The *Streaming* component is one of 360Anywhere's underlying components. In our case, streaming does not only refer to broadcasting the 360-degree video feed (and optional Audio/Video Chat), but also includes synchronization of annotations between all collaborators as well as between the Remote and Projection UIs. For synchronizing video streams across collaborators, we build on a

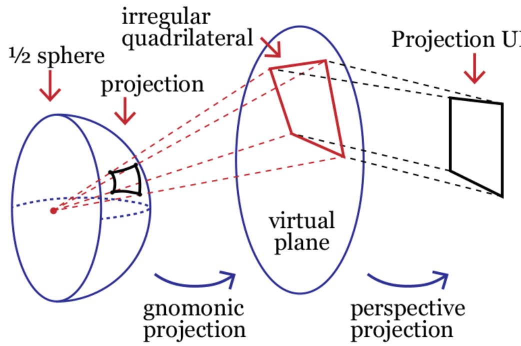

Fig. 4. Transformation process between a projection on the 360 sphere and a Projection UI using gnomonic projection and perspective projection.

dedicated *WebRTC* library that has been specifically developed for 360Anywhere and enables many-to-many streaming in different channels (one channel for the 360-degree stream, one channel for Audio/Video Chat). This component provides additional web socket functionality for synchronizing non-video/non-audio content in real time, which is used for Annotations, Gaze Awareness, and Follow Me.

To detect and resolve synchronization issues, it is possible to use Audio/Video Chat (in case only the 360-degree video stream is affected), or automatically listen for Socket.IO `disconnect` events.

4.2 Annotations & Collaborator UI

The *Annotations* component is the other underlying component of 360Anywhere. It enables users to draw, place images, and place text directly into the 360-degree stream in the Collaborator UI. The video feed is used as the inner texture of a sphere in an *A-frame* scene, with the camera being positioned at the center of that sphere. When drawing, using *three.js*, lines are placed directly on the curved surface of the sphere based on a ray casting approach. For simplicity, in our implementation, rather than lying flat on the curved surface, images remain planes whose four corners touch the sphere. The visual appeal of this solution is almost the same as with the more complex approach, but with less overhead. Text to be added to the scene is entered into a regular input field, the content of which is then transformed into an image using an HTML5 *canvas* element. In addition to their global coordinates within the 360-degree stream, all annotations have a *sphere index*, which is analogous to a CSS *z-index*. Therefore, it determines the order of annotations that overlay each other. The sphere index is determined by the order in which annotations are added to the scene. If an annotation is moved after it has been inserted, it is automatically moved to the front. To cater for mobile scenarios with touch input, the scene is frozen when the user draws or moves an annotation using one of their fingers (cf. [3, 5, 12]).

4.3 Projection

Projection is made possible through the *Calibration* and *Transformation* components. First, when the user calibrates a projection in the Collaborator UI, they have to click the four points on the sphere that correspond to the four corners of the projection. Once one or more projections have been calibrated, the Transformation component takes care of propagating all annotations that lie within the boundaries of a projection to the correct Projection UI (Figure 4). For this, in a first step the sphere is mapped to virtual planes using *gnomonic projection* (separately for both hemispheres). Gnomonic projection is preferred over cylindrical projection in this case because it reverses the fisheye lens distortion algorithm of the 360-degree camera, which significantly simplifies the image processing procedure on the client and reduces deviation. That is, using cylindrical projection, the mapping would be increasingly inaccurate the farther away the projection is from the sphere's equator, which is undesirable. In a second step, since we know that the Projection UI is rectangular and we know the four points defining the projection on the virtual plane, we can create a mapping and calculate the corresponding transformation matrix. Using this matrix and after transforming their global to local coordinates, all annotations within the projection can then be displayed at their correct positions in the Projection UI.

5 EVALUATION

To evaluate 360Anywhere we conducted an exploratory user study in the form of three design jams. For this, we presented the participants with pre-defined scenarios and asked them to develop and reason respective 360Anywhere configurations while enacting the scenarios. Moreover, we elicited additional use cases in which the participants deemed our framework useful. The goal of this evaluation was to investigate:

(1) whether the different components of 360Anywhere can mitigate the challenges identified by Tang *et al.* [23];
(2) whether 360Anywhere is feasible in mobile, ad-hoc scenarios; and
(3) whether 360Anywhere is flexible enough to be applied to a variety of different scenarios.

Our results suggest that 360Anywhere is able to provide all of this.

5.1 Scenarios

In the following we introduce the scenarios used in our study. They were carefully selected to represent realistic and relevant use cases and cover different configurations of 360Anywhere. In particular, they were chosen to

(1) cover 1-to-1, 1-to-many, and many-to-many interaction;
(2) provoke all of the identified challenges with 360-degree video collaboration;
(3) ensure that each of 360Anywhere's components was required at least once; and
(4) provide increasing complexity and decreasing feasibility of using other systems (e.g., while Scenario #1 uses one wall, Scenario #3 requires interaction on two walls with projective AR and actively looking around in the 360-degree live feed).

For each scenario, we hypothesize the set of framework components required to successfully support it. We also identify components that could be included, but are not crucial to support a given scenario. It is important to note that the scenarios were not chosen to assess the advantages of 360-degree video over traditional video conferencing since these have already been described by [23].

5.1.1 Baseline: 360-degree Video Collaboration. The baseline configuration of 360Anywhere includes Streaming and Annotations, but no optional components or projections. In this configuration, the framework supports a basic 360-degree video collaboration use case—even if the collaborators do not make use of annotations. Since this use case has already been studied in detail by Tang *et al.* [23], it will *not* be included in our evaluation.

5.1.2 Scenario #1: Meeting with Remote Participants. The interaction design team of a medium-sized design furniture company meets with two remote freelancers to brainstorm the first version of the mobile UI of their new online shop. In particular, they want to discuss and sketch out potential visual layouts and menu structures. The local set-up at the company office is a meeting room equipped with a 360-degree camera, a projector and a writeable wall that acts as a whiteboard. The remote participants of the meeting use their own devices to participate. One freelancer uses a laptop, the other a smartphone.

This is a *many-to-many* scenario using *one projection*. Required components to support the scenario are: (a) *Audio/Video Chat*, because everyone needs to talk to everyone else, and (b) *Session Support*, because the results of such a meeting must be available for later review and implementation. *Follow Me* is not required since the scenario has a limited number of participants without a single instructor. Hence, coordination can instead easily happen through Audio/Video Chat. Optional components are: *Gaze Awareness*, since participants can articulate at which icon they are looking as well, and *Back in Time*, since one can alternatively ask other collaborators to repeat a statement they missed.

5.1.3 Scenario #2: Online Lecture. An archaeology professor gives an online lecture about a dig in Egypt that she is currently visiting. She carries a laptop (including a webcam) as well as a 360-degree camera with her for broadcasting the dig site during the lecture. The lecture can be accessed through a website by all students enrolled in her course about ancient Egypt.

This is a *1-to-many* scenario. Required components to support the scenario are: (a) *Follow Me*, because the lecturer must be able to draw students' attention to details they are talking about, and (b) *Session Support*, because the lessons learned must be available to students for exam preparation or similar. *Audio/Video Chat* should not be included in this scenario due to the potentially very large number of remote collaborators that would clutter the channel. The same holds for *Gaze Awareness*. An optional component that could be included is *Back in Time*, in case students miss something the lecturer explained. However, if a student uses Back in Time they might miss what is currently happening in the live stream.

5.1.4 Scenario #3: Construction Site. A project manager needs to give short-term instructions on what to do next on a construction site in San Francisco he is responsible for. However, he is currently on a business trip in Japan and therefore cannot give the instructions on site. In particular, his input is needed to determine the next steps concerning drilling and cutting work on two walls in a particular room. That is, it must be defined where to cut into the walls and where not to drill to avoid electric wires and gas pipes. The local set-up prepared by the three workers responsible for the room is a 360-degree camera on a tripod as well as two projectors projecting onto the two walls. The project manager provides input using his laptop.

This is a *many-to-one* scenario using *two projections*. Required components to support the scenario are: (a) *Audio/Video Chat*, because bidirectional communication is necessary, and (b) *Session Support* for later implementation of the project manager's directions. *Back in Time* is not required in this scenario since the only remote collaborator is also the person giving instructions. In case the local workers miss something, coordination should happen through Audio/Video Chat. Optional components are: *Gaze Awareness*, because it should be clear where the project manager is looking when he draws on a wall, and *Follow Me*, because the local workers might want to draw the managers attention to certain parts of the room. However, such coordination can also happen through Audio/Video Chat.

5.2 User Study

The user study was carried out in terms of three design jams on different days, each with a different team of participants. The sessions were meant to be exploratory: although we provided the teams with pre-defined tasks to be solved, our main aim was to gain a better understanding of how 360Anywhere and its individual components are used *in situ*. We wanted to observe how users design a remote collaboration system based on the framework for specific use cases and determine in which other scenarios they would deem it useful.

Fig. 5. The design jam set-up: (1) an Acer Predator Notebook connected to (2) a RICOH Theta S 360-degree camera for the local participants; (3) a projector connected to a Microsoft SurfaceBook; (4) icons provided for scenario #1; (5) another Acer Predator Notebook for one remote collaborator; (6) an ASUS ZenFone AR for the other remote collaborator; (7) a 4K screen acting as the second projection in scenario #3; and (8) hieroglyphic symbols for scenario #2.

So far, there is no comparable system that integrates 360-degree video, remote annotations, and augmented reality. Therefore, we do not want to restrict ourselves to comparisons with systems that are limited to just one of the related domains (e.g., telepresence), since we aim at covering a broader range of use cases. Furthermore, because of the general complexity of the framework and the contained technology, we first need to get a better understanding of the possibilities of 360Anywhere as well as how and for which use cases collaborators would use it. Due to this, we have chosen to investigate our approach in exploratory settings. The set-up and apparatus for the design jams are given in Figure 5.

5.2.1 Participants. The participants of the user study were interaction design Master's students recruited from the University of Michigan School of Information. Overall, we were able to recruit 10 students (average age = 23.0, 2 male), with backgrounds ranging from design to engineering to business, for the three sessions, i.e., two groups of three and one group of four. Participants were fairly familiar with 360-degree content (median = 2 on a 5-point scale), moderately familiar with augmented reality (median = 3), familiar with remote collaboration (median = 4; mostly Google Docs, Skype, Dropbox, and WebEx), and very familiar with audio/video chat (median = 5).

Subsequently, the group of participants went on to take part in the actual study.

5.2.2 Procedure. The design jams took place in a flexible meeting room with whiteboard walls. The room configuration was changed to simulate each scenario. Participants acting as remote collaborators in the scenarios were sent to a separate room. We first briefed participants (5 min.), provided them with a printed manual explaining how to set up and use the framework, and gave a detailed walkthrough. This included selecting cameras, setting and calibrating projections, choosing configurations, broadcasting to remote collaborators, and also explanation of the usage of each component. This walkthrough, guided by the printed manual, took between 30 and 60 minutes, depending on how many questions the participants asked. During this, participants also familiarized themselves with the system by trying different configurations and functionalities.

Once the participants felt they were reasonably familiar with 360Anywhere, they were presented with a written description of the scenarios mentioned above, which also included specific tasks and necessary materials: In scenario #1, the local collaborator should prepare a selection of icons and together with the remote collaborators decide on one for "payment" and one for "feedback" (cf. Figure 5, ④). In scenario #2, the lecturer should use a table with hieroglyphic symbols to teach remote students the corresponding Latin characters for three symbols (cf. Figure 5, ⑧). In scenario #3, the local workers should mark two points on one wall while the remote project manager should point out how to connect them with an electric wire. Additionally, the manager should mark three safe areas for drilling on another wall (cf. Figure 5, ⑦). The scenarios could be completed in any order and all three scenarios took teams between 55 and 75 min.

Before each scenario, the team discussed and filled out a questionnaire specifying which configuration they chose. For each component, they had to provide the specific reasons why they believed it should or should not be included for the scenario. Next, one or two participants (depending on the scenario) acting as the local collaborators set up the configuration independently, and the other one or two participants went outside the room with the devices for the remote collaborators. The URL for the remote collaborators provided by the 360Anywhere Configuration was sent by the local collaborators through a dedicated Slack channel. During this process, three experimenters recorded the time for setting up, observed participants, and noted down the use of the system and its components, participants' comments, and technical limitations of the system. After each scenario, the team discussed whether they would adjust their configuration (by adding or removing components based on the gained experience) and if so, in which way. The same process was applied for the remaining two scenarios, with participants switching roles between local and remote collaborators.[1]

After all scenarios were finished, we asked participants to imagine at least one other scenario in which they would deem 360Anywhere useful, and to explain how they would make use of the framework. This was followed by a post-study questionnaire including demographic questions, a subjective assessment of the framework based on Likert scales, and questions about its main benefits and limitations (10 min.) Completing the walkthrough and three scenarios with one team took between 1h 40min and 2h 30min, depending on the amount of questions and feedback.

5.3 Performance

The set-up of the design jams was based on *Google Chrome* v61.0.3163, which supports the required web technologies: A-Frame, Socket.IO, and WebRTC. In our tests, we achieved good performance on mini-PCs and laptops (NUC, ThinkPad), as well as Android phones (ZenFone, Nexus 5, Pixel 2), which are everyday mobile devices and would be common in the targeted scenarios. We found that the number of active streams has the biggest performance impact and therefore Back in Time (which essentially duplicates the 360-degree stream) and Audio/Video Chat constitute bottlenecks with a growing number of collaborators. This became evident in scenarios with four participants and both components active. In such cases, it would be possible to automatically reduce the resolution of the WebRTC stream(s) and/or disable certain components.

	Scenario 1				Scenario 2				Scenario 3		
	hypothesis	T1	T2	T3	hypothesis	T1	T2	T3	hypothesis	T2	T3
Audio/Video Chat	✔	✔	✔	✔		✔		✔	✔	✔	✔
Gaze Awareness	(✔)	(✔)	(✔)				✔		(✔)		
Follow Me		✻		✻	✔	✻	✔	✔	(✔)		✔
Sessions	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
Back in Time	(✔)	(✻)			(✔)	✻	✔	✔			

Table 1. The teams' configuration choices compared to our hypothetical optimal configurations. ✔ = component included in configuration and used during scenario; ✻ = component included but not used during scenario; (•) = component optional or included "just to be sure".

5.4 Results

In the following, we report on the results of the three design jams. First, by looking at each component individually, we assess whether participants used it correctly for their design and whether it contributed to solving existing challenges. Then, we summarize other observations and feedback before describing 360Anywhere's benefits and limitations, as per the participants' questionnaire responses.

5.4.1 Components. Participants largely confirmed our anticipated configurations for the scenarios (Table 1). That is, all of the components we considered necessary for each scenario were included by all teams. Yet, at least initially, participants showed the tendency to include more components than we thought were necessary. In five cases, teams included a component that we did not think required, three of which eventually decided to remove it from the configuration after trying it out in the given scenario. In seven cases, teams included a component that we considered to be optional, two of which were removed from the configuration after testing it in the scenario.

Audio/Video Chat was included seven times across all scenarios and teams. Reasons given by the teams for including this component were to facilitate bidirectional audio-visual discussion (5×) and the possibilities to ask questions and see the instructor in a 1-to-many set-up. The reason given for excluding the component in one case was that students should ask questions only after an online lecture. One team removed Audio/Video Chat again after S2, stating that lots of students would clutter the 360-degree stream. In general, all teams made active use of the component whenever it was included. On two occasions the chat was deactivated during a scenario because it was covering important parts of the stream (T1S1). One team stated there is no need for Audio/Video Chat in 1-to-many scenarios (since the 360-degree camera is also broadcasting audio); two teams suggested a specific 1-to-many chat for such cases.

In general, this could indicate that Audio/Video Chat is used in 1-to-few and few-to-few, but not in 1-to-many or many-to-many scenarios, as this would clutter the screen.

Gaze Awareness was included in a scenario three times, the reasons being "just in case" (2×) and as a control mechanism for the lecturer in S2. On five occasions, the component was not included. Three times the reason was that it is simply not necessary (e.g., , "It doesn't matter where people are looking at," T3S1). Also, one team stated that collaborators know where someone is looking anyways when drawing is active; another team said the cones would clutter the stream. Two teams removed Gaze Awareness again after the scenario because the component did not provide added value (T1S1, T2S2). During the scenarios, participants from two teams said the component feels strange or distracting and suggested less jarring cones. One of them also noticed potential privacy issues ("Can I hide what I'm looking at?"). One participant mentioned there is no

[1]Due to time constraints, the first team completed only scenarios #1 and #2, in a session of 2.5 hours.

need for Gaze Awareness when the Follow Me component is active (T1S2) while another said it is not necessary when you have audio since everyone can say what they are looking at (T1S1). In one case, a team discussed intensively that Gaze Awareness might just be distracting, but none of the members deactivated the component during the following scenario and they did not decide to remove it afterwards (T2S1). During S3, one participant stated "I can't see any scenario where Gaze Awareness makes sense," but later they acknowledged it might be useful when there are multiple areas of interest. Furthermore, one team inquired into whether Gaze Awareness tracks a collaborator's eyes or just their viewport (which is the case), and suggested that the former would be much more useful.

In general, this could indicate that Gaze Awareness is not necessary with few participants when annotations are used. However, participants noted that this component is not designed optimally and its functionality is partly covered by other components, which requires further investigation.

Follow Me was included six times across all teams and scenarios. Reasons for inclusion were to actively direct other collaborators' attention in a bidirectional scenario (3×) and to enable an instructor to control focus in 1-to-many scenarios (3×). Twice, teams saw no need to include the component because there were only one or two areas of interest present in the scenario. One team removed the component after S1, stating there was no need for it with a single area of interest. In three cases in which Follow Me was included, it was not used during the scenario (T1S1, T1S2, T3S1). Two remote collaborators explicitly complained about this afterwards ("You didn't use Follow Me!" – "Ooh, I didn't notice!", T3S2). In one of the cases, a local collaborator noted that they thought the others would share their view anyway, i.e., there was a lack of awareness that all collaborators can control the 360-degree stream independently. One team stated that Follow Me is not necessary when Audio/Video Chat is present since everyone can say what they are looking at (T1S1).

In general, this could indicate that Follow Me is used in scenarios with an instructor, but not in scenarios with only a few areas of interest.

Session Support was included in all scenarios by all teams, five times to enable later review and three times to serve as an input for later steps (in S1 and S3). During S2, one team explicitly mentioned the usefulness of this component. All teams intended to make use of the feature to persist the digital artifacts they produced during the scenarios. However, in four cases (T2S1, T1S2, T2S3, T3S3) a remote collaborator cleared all annotations—either by accident or because they were playing around with the different features—before they could do so.

In general, this could indicate that Session Support is useful independent of the scenario.

Back in Time was included four times across all teams and scenarios. Reasons given for inclusion were to be able to catch up with points collaborators have missed (3×), or as a safeguard ("just in case"). When not included, the common reason was that collaborators could just as well ask others to repeat something, i.e., if Audio/Video Chat is active. In one case a team removed it after S2, the reason being that Back in Time is generally useful, but as the collaborator uses it, they may miss other things happening in the live stream. The solution proposed by the team was to provide a split view that shows the current live stream in a smaller overlay while using Back in Time. Another team explicitly stated that the component is helpful for the students in S2. On two occasions, Back in Time was included in the system, but not used (T1S1, T1S2).

In general, this could indicate that Back in Time is useful in 1-to-many scenarios, if interrupting to ask collaborators is not desired or not possible, e.g., if Audio/Video Chat is not available.

5.4.2 Other Observations & Feedback. In this section, we report on other observations and feedback not directly related to one of the components above, including feedback from the post-study questionnaire.

Fig. 6. Example 360-degree snapshot persisted by the Session component (T1S1). The inlay shows how T1 approached explaining the hieroglyphic symbols in scenario #2.

	median	range
Supports a range of different scenarios	4.5	[3, 5]
Would be effective in other scenarios	4	[4, 5]
Felt easy to configure the system	4	[2, 5]
Felt fast to configure the system	4	[1, 5]
Enjoyed using the system	4	[3, 5]
Has advantages over existing solutions	4	[3, 4]
Easy adaptation to different scenarios	3.5	[2, 5]
Was effective in studied scenarios	3	[3, 5]
Felt easy to use the system	3	[2, 4]
System is flexible	3	[1, 5]

Table 2. Participants' ratings for 10 statements about the framework on a 5-point Likert scale.

The common strategy for *Scenario #1* was to place the print-outs of the icons that were provided to the teams on the wall that was enhanced with the projection. The local collaborator marked their favorite icons using drawing mode (either by circling or drawing numbers) and instructed the remote collaborators to provide their feedback. These then also marked their preferred choices using the drawing feature and decided on the winners based on majority vote (cf. Figure 6). T2 and T3 additionally placed check marks next to the two icons they finally decided on. In this scenario, three participants did not make use of the possibility to look around in 360 degrees at all. Judging by the collected feedback this was due to the fact that there was only one area of interest present that the viewport fully covered. This scenario was successfully completed by all teams.

The common strategy to tackle *Scenario #2* was that the "lecturer" first explained what they were going to do, making use of the 1-to-many broadcasting of the 360-degree camera. Subsequently, one after the other, they would circle the three symbols they should explain (F, J, and M) and place the corresponding latin character next to it using 360Anywhere's text feature (cf. Figure 6, inlay). One team circled the three symbols in different colors with the latin characters in the respective color, in order to make the mapping clearer for the "students". On two occasions, participants mentioned that students should not be able to draw into the feed. This scenario was successfully completed by all teams.

In *Scenario #3*, all of the teams had the remote "project manager" point out the three safe areas on the one wall using drawing mode (comparable to Figure 5, ⑦). Subsequently, the local "workers" defined two points A and B on the other wall. For this, both teams (T2 and T3) chose to rely on 360Anywhere and draw the points digitally into the 360-degree stream rather than using whiteboard markers, although they were informed that they could write on the IdeaPaint dry-erase wall. During the whole process, the "project manager" used the 360-degree live stream to actively look around for interacting with the different walls. This scenario was successfully completed by all teams.

In general, although the framework seemed highly complex to some participants during the walkthroughs ("I'm not sure I understand ... anything, yet"), all teams were able to successfully complete all scenarios. For each scenario, the teams decided on the components to be included, prepared the corresponding configuration in the Configuration UI, and began the scenario in under 10 minutes.

Several teams suggested certain features to add to 360Anywhere. One stated that they missed a shared document or a chat function that would be separate from the stream, in order to take notes and be able to send messages to the collaborators without having to write directly into the 360-degree stream. Another team mentioned that an undo function would be highly useful, because currently, only *all* annotations can be cleared. Also, after one collaborator accidentally cleared the whole scene before the team could store the session, an individual clear function was suggested. Finally, a zoom function was proposed by two teams, both times w.r.t. recognizing the hieroglyphic symbols in S2.

In the post-study questionnaire, we asked participants to rate 10 statements about 360Anywhere based on a 5-point Likert scale from *strongly disagree* to *strongly agree* (Table 2). They agreed that the system is easy and fast to configure and enjoyed using it during the design jams ("I enjoyed the process."). Yet, they felt only neutral about ease of use, which is understandable given the complexity of the involved technology. In general, they rated flexibility and effectiveness in the studied scenarios as being average. For all of the the remaining statements, 360Anywhere was rated above average. Particularly, participants acknowledged its potential for supporting and being effective in different scenarios. Also, they generally stated that our framework has advantages over existing solutions for remote collaboration.

5.4.3 Benefits. In the post-study questionnaire, we also asked participants to state what they considered to be the main benefits of our solution. Four mentioned the easy access to the local environment and the possibility to freely look around in the 360-degree stream while another four stated that 360Anywhere provides more possibilities for interaction than in traditional settings. Furthermore, two participants considered the possibility to draw on existing surroundings (i.e., projective AR) as the main benefit. Another two mentioned the variety of scenarios and flexibility of the framework. Gaze Awareness, the possibility to see the people you're interacting with ("no one is slacking off because you see everyone"), easier communication and understanding, low latency, and more engaging interaction were each mentioned once.

5.4.4 Limitations. We also asked participants to name the main limitations of 360Anywhere. For four of them, these were the technical limitations of the prototype (e.g., the ZenFone crashed occasionally and had to be reloaded, calibration could not be aborted, and the system got stuck and had to be reloaded in some cases). Two participants noted that the framework was rather complex and confusing while another two stated it was easy to lose focus (due to the 360-degree view) and

360Anywhere would not be effective in scenarios that need focus. A lack of flexibility and the fact that there is no chat separate from the 360-degree stream were mentioned once each.

5.4.5 Other scenarios. At the end of each design jam, we elicited at least one scenario from the respective team in which they would deem 360Anywhere useful and asked specifically how they would make use of our framework. The following is what the participants came up with.

PowerPoint integration. 360Anywhere could be combined with PowerPoint in a way that lets local collaborators integrate a slideshow into a projection, which can then be digitally annotated and persisted by remote collaborators. This use case would be highly useful in discovery sessions with clients. A particular challenge to solve in order to make this use case feasible are different annotations for the different slides contained in the presentation.

Surgery. Using projective AR on the body of a patient as well as a high-resolution 360-degree camera, 360Anywhere could be used for surgery. That is, a remote doctor acts as a consultant to a team of local surgeons, thereby drawing instructions or similar directly on the patient's body.

Teaching in medicine. 360Anywhere could be used to project organs, muscles, or bones, among other things, onto the model of a human body. A lecturer in medicine could in this way hold an online lecture for remote students while the model of the body serves as an interactive surface onto which the lecturer can draw digital annotations. This is similar to *Augmented Studio* by [7], however in a more low-fidelity scenario with less complex hardware requirements and set-up.

Crime scene investigation. To enable remote consultants (e.g., detectives, forensic specialists) to investigate a crime scene almost as if they were there, 360Anywhere could be set up on site. This—like in the surgery use case—would require a high-res 360-degree camera as well as a small, easily portable projector. In this way, the remote expert could highlight points of interest in the crime scene that would require closer attention by the local crew.

5.4.6 Findings. Overall, study participants were very appreciative of 360Anywhere's ability to support a variety of collaboration scenarios ("This is obviously useful for collaboration"). They believed our solution could be adapted to and also be effective in a range of different use cases, despite being more critical about effectiveness and ease of use in the studied scenarios—technical issues were mentioned as the framework's main limitation. Also, the fact that students were generally able to set up and use a remote collaboration system using our framework in less than 10 minutes shows its feasibility for mobile, ad-hoc scenarios. Other parts of the feedback we received were rather heterogeneous. To give just one example, ratings for the statement "The system is flexible" ranged from *strongly disagree* to *strongly agree*, with a median of 3, which illustrates the complexity of the approach and users' uncertainty about its scope. Furthermore, flexibility was mentioned as both, a main benefit and main limitation by the participants.

The different components of the framework were mostly used in the ways we anticipated. While the usefulness of being able to persist digital artifacts in a room *(C5: Persistence)* and therefore Session Support were undisputed—it was included in all scenarios—participants were unsure about the usefulness of Gaze Awareness, which was included in only three of eight cases. However, they generally acknowledged the necessity to be able to see where collaborators are looking *(C1: Gaze)* and suggested Audio/Video Chat, Follow Me, and Annotations already provide a solution for this. Audio/Video Chat was moreover appreciated by the collaborators because it facilitated bidirectional audio-visual discussion, i.e., they wanted to see the other collaborators and what they were doing *(C3: Gestures)*. Follow Me was deemed useful in 1-to-many scenarios with a single instructor and participants recognized the need for synchronizing views to direct other collaborators' attention *(C2: Out-of-sync)*. Yet, they also stated that the component is not necessarily required in scenarios with only one area of interest or could be replaced by Audio/Video Chat. Finally, participants acknowledged the need to be able to catch up on important points one missed in a collaborative scenario *(C4: Attention)*. They stated that the Back in Time component can solve this problem if Audio/Video Chat is not present, but requires technical adjustments to make sure collaborators do not miss anything else in the 360-degree live stream.

To conclude, according to our study participants,

C1: Gaze can be solved by Audio/Video Chat, Gaze Awareness, Follow Me, and Annotations;
C2: Out-of-sync by Audio/Video Chat and Follow Me;
C3: Gestures by Audio/Video Chat;
C4: Attention can be solved by Audio/Video Chat and also could be solved by Back in Time (adjustments required); and ultimately,
C5: Persistence can be solved by the Session Support component.

6 CONCLUSION

We present 360Anywhere, a framework for mobile ad-hoc remote collaboration in augmented reality, based on 360-degree live input. We enable users to quickly and easily define remote collaboration systems that are tailored to their needs, being able to choose from a range of components. Through a projective AR set-up, remote collaborators are able to add digital annotations to a real-world environment by drawing directly into a 360-degree video stream. In three design jams with interaction design Master's students, we have investigated the effectiveness, feasibility, and potential of our solution. Particularly, we wanted to gain a better understanding of how collaborators configure the system for different scenarios and how the chosen components are used *in situ*. Results show that 360Anywhere is able to solve existing challenges regarding 360-degree video collaboration that were identified in prior work. Given the fact that participants in the user studies were able to configure and start using a system based on 360Anywhere in under 10 minutes and all teams were able to complete all scenarios, our approach seems to have a low *threshold* [16] given the complexity of the involved technology. Moreover, 360Anywhere has a high *ceiling* [16], with a total of 64 possible combinations of 6 selectable components, which cover a wide range of scenarios, as has been also acknowledged by the study participants. Finally, as a systems contribution, 360Anywhere addresses unresolved challenges, reduces solution viscosity, empowers new design participants, and leverages power in combination (cf. [17]).

Potential directions for future work include automatic calibration of the projective AR part of the system (similar to RoomAlive [10]), revising Gaze Awareness, and providing alternatives to Follow Me, similar to the solutions presented in [14]. Moreover, it could be interesting to investigate integration with HoloLens so that collaborators see digital annotations through a head-worn display rather than relying on projections. However, this would raise a number of new technical challenges, in particular, when it comes to calibration and synchronization of 360-degree remote annotations.

REFERENCES

[1] Tomoaki Adachi, Takefumi Ogawa, Kiyoshi Kiyokawa, and Haruo Takemura. 2005. A telepresence system by using live video projection of wearable camera onto a 3d scene model. In *Proc. HCII*.

[2] Carolina Cruz-Neira, Daniel J. Sandin, and Thomas A. DeFanti. 1993. Surround-screen Projection-based Virtual Reality: The Design and Implementation of the CAVE. In *Proc. SIGGRAPH*. 135–142. https://doi.org/10.1145/166117.166134

[3] Omid Fakourfar, Kevin Ta, Richard Tang, Scott Bateman, and Anthony Tang. 2016. Stabilized Annotations for Mobile Remote Assistance. In *Proc. CHI*. 1548–1560. http://doi.acm.org/10.1145/2858036.2858171

[4] Andreas Rene Fender, Hrvoje Benko, and Andy Wilson. 2017. MeetAlive: Room-Scale Omni-Directional Display System for Multi-User Content and Control Sharing. In *Proc. ISS*. 106–115. http://doi.acm.org/10.1145/3132272.3134117

[5] Steffen Gauglitz, Benjamin Nuernberger, Matthew Turk, and Tobias Höllerer. 2014. World-stabilized Annotations and Virtual Scene Navigation for Remote Collaboration. In *Proc. UIST*. 449–459. http://doi.acm.org/10.1145/2642918.2647372

[6] Michael Haller, Jakob Leitner, Thomas Seifried, James R. Wallace, Stacey D. Scott, Christoph Richter, Peter Brandl, Adam Gokcezade, and Seth E. Hunter. 2010. The NICE discussion room: integrating paper and digital media to support co-located group meetings. In *Proc. CHI*. 609–618. http://doi.acm.org/10.1145/1753326.1753418

[7] Thuong Hoang, Martin Reinoso, Zaher Joukhadar, Frank Vetere, and David Kelly. 2017. Augmented Studio: Projection Mapping on Moving Body for Physiotherapy Education. In *Proc. CHI*. 1419–1430. http://doi.acm.org/10.1145/3025453.3025860
[8] Brad Johanson, Armando Fox, and Terry Winograd. 2002. The Interactive Workspaces Project: Experiences with Ubiquitous Computing Rooms. *IEEE Pervasive Computing* 1, 2 (2002), 67–74. https://doi.org/10.1109/MPRV.2002.1012339
[9] Brett R. Jones, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. 2013. IllumiRoom: Peripheral Projected Illusions for Interactive Experiences. In *Proc. CHI*. 869–878. https://doi.org/10.1145/2470654.2466112
[10] Brett R. Jones, Rajinder Sodhi, Michael Murdock, Ravish Mehra, Hrvoje Benko, Andrew Wilson, Eyal Ofek, Blair MacIntyre, Nikunj Raghuvanshi, and Lior Shapira. 2014. RoomAlive: magical experiences enabled by scalable, adaptive projector-camera units. In *Proc. UIST*. 637–644. https://doi.org/10.1145/2642918.2647383
[11] Hannes Kaufmann and Dieter Schmalstieg. 2003. Mathematics and geometry education with collaborative augmented reality. *Computers & Graphics* 27, 3 (2003), 339–345. https://doi.org/10.1016/S0097-8493(03)00028-1
[12] Seungwon Kim, Gun A. Lee, Sangtae Ha, Nobuchika Sakata, and Mark Billinghurst. 2015. Automatically Freezing Live Video for Annotation During Remote Collaboration. In *CHI Extended Abstracts*. 1669–1674. http://doi.acm.org/10.1145/2702613.2732838
[13] Jerry Li, Mia Manavalan, Sarah D'Angelo, and Darren Gergle. 2016. Designing Shared Gaze Awareness for Remote Collaboration. In *Proc. CSCW Companion*. 325–328. https://doi.org/10.1145/2818052.2869097
[14] Yen-Chen Lin, Yung-Ju Chang, Hou-Ning Hu, Hsien-Tzu Cheng, Chi-Wen Huang, and Min Sun. 2017. Tell Me Where to Look: Investigating Ways for Assisting Focus in 360° Video. In *Proc. CHI*. 2535–2545. http://doi.acm.org/10.1145/3025453.3025757
[15] Microsoft. 2018. Skype on HoloLens. (2018). https://www.microsoft.com/en-us/hololens/apps/skype.
[16] Brad Myers, Scott E. Hudson, and Randy Pausch. 2000. Past, Present, and Future of User Interface Software Tools. *ACM Trans. Comput.-Hum. Interact.* 7, 1 (March 2000), 3–28. https://doi.org/10.1145/344949.344959
[17] Dan R Olsen Jr. 2007. Evaluating User Interface Systems Research. In *Proc. UIST*. 251–258. http://doi.acm.org/10.1145/1294211.1294256
[18] Jun Rekimoto and Masanori Saitoh. 1999. Augmented Surfaces: A Spatially Continuous Work Space for Hybrid Computing Environments. In *Proc. CHI*. 378–385. https://doi.org/10.1145/302979.303113
[19] Rajinder S. Sodhi, Brett R. Jones, David Forsyth, Brian P. Bailey, and Giuliano Maciocci. 2013. BeThere: 3D Mobile Collaboration with Spatial Input. In *Proc. CHI*. 179–188. https://doi.org/10.1145/2470654.2470679
[20] Norbert A. Streitz, Jörg Geißler, Torsten Holmer, Shin'ichi Konomi, Christian Müller-Tomfelde, Wolfgang Reischl, Petra Rexroth, Peter Seitz, and Ralf Steinmetz. 1999. i-LAND: An Interactive Landscape for Creativity and Innovation. In *Proc. CHI*. 120–127. https://doi.org/10.1145/302979.303010
[21] Mengu Sukan, Carmine Elvezio, Ohan Oda, Steven Feiner, and Barbara Tversky. 2014. ParaFrustum: Visualization Techniques for Guiding a User to a Constrained Set of Viewing Positions and Orientations. In *Proc. UIST*. 331–340. https://doi.org/10.1145/2642918.2647417
[22] Anthony Tang and Omid Fakourfar. 2017. Watching 360° Videos Together. In *Proc. CHI*. 4501–4506. http://doi.acm.org/10.1145/3025453.3025519
[23] Anthony Tang, Omid Fakourfar, Carman Neustaedter, and Scott Bateman. 2017. Collaboration in 360° Videochat: Challenges and Opportunities. In *Proc. DIS*. 1327–1339. http://doi.acm.org/10.1145/3064663.3064707
[24] Vuforia. 2018. Project Chalk. (2018). https://chalk.vuforia.com/.
[25] Andrew D. Wilson and Hrvoje Benko. 2010. Combining Multiple Depth Cameras and Projectors for Interactions on, Above and Between Surfaces. In *Proc. UIST*. 273–282. https://doi.org/10.1145/1866029.1866073

Received February 2018; revised April 2018; accepted June 2018