General-Purpose Telepresence with Head-Worn Optical See-Through Displays and Projector-Based Lighting

Andrew Maimone* Xubo Yang† Nate Dierk* Andrei State* Mingsong Dou* Henry Fuchs*

*University of North Carolina at Chapel Hill †Shanghai Jiao Tong University

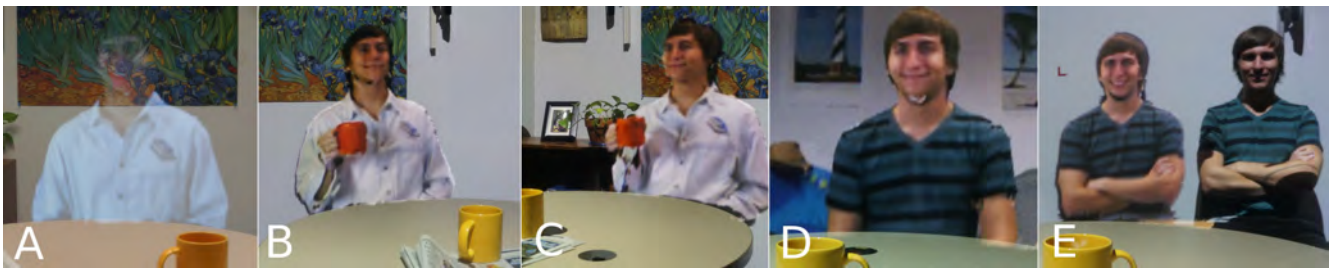

Figure 1: Left to right: A) Without lighting control, virtual imagery shown through a conventional optical see-through display appears transparent and low contrast in a normally lit room. B-C) Remote user appears *inside* local environment from different tracked viewing positions with projector-based lighting control. D) Remote user and his environment appears as an *extension* of the local environment. E) Participant sitting next to his (paused) virtual representation. All images show a live tracked telepresence session filmed through a head-worn optical see-through display.

ABSTRACT

In this paper we propose a *general-purpose* telepresence system design that can be adapted to a wide range of scenarios and present a framework for a proof-of-concept prototype. The prototype system allows users to see remote participants and their surroundings merged into the local environment through the use of an optical see-through head-worn display. Real-time 3D acquisition and head tracking allows the remote imagery to be seen from the correct point of view and with proper occlusion. A projector-based lighting control system permits the remote imagery to appear bright and opaque even in a lit room. Immersion can be adjusted across the VR continuum. Our approach relies only on commodity hardware; we also experiment with wider field of view custom displays.

Keywords: teleconferencing, augmented reality, virtual reality

Index Terms: H.4.3 [Information Systems Applications]: Communications Applications—Computer conferencing, teleconferencing, and videoconferencing

1 INTRODUCTION

The goal of telepresence is to create the feeling that one is present in a remote place or co-located with a remote person. In the visual sense, past systems have created this illusion through a variety of paradigms, among them: a remote user appearing *inside* the local environment [4], a remote space appearing to *extend* beyond the local environment [2, 8], and a local user *immersed* in a remote place [3].

Of course the most appropriate telepresence paradigm depends on context and may change frequently within a gathering. As an illustrative example, we imagine that an architect, located in a remote studio, addresses clients located in a meeting room. The architect first describes why a new building might benefit the clients. The clients see him sitting at an empty seat in the room and feel that he is among them, establishing trust. Next, the architect shows models of the building that he has constructed in his studio. The clients now see the architect's studio extending from one of the walls of the meeting room – they look around to assess the models while also gauging their local colleagues interest. Finally, the architect shows them his centerpiece – the lobby. The clients are now completely immersed in the lobby of the building and each is free to inspect different aspects of the design.

In the above example, we described a flexible telepresence system that continuously adapts to the most appropriate immersion mode depending on the situation. In this paper, we define the requirements for such a *general-purpose* telepresence system (in the visual sense) and suggest how one could be built. We also provide a framework used to build a limited prototype that is based on optical see-through head-worn displays and projector-based lighting control. The prototype allows users to see remote participants and their environments with proper mutual occlusion and precise control over the level of immersion.

*e-mail: {maimone,nate,andrei, doums, fuchs}@cs.unc.edu

†e-mail: yangxubo@sjtu.edu.cn

2 BACKGROUND AND CONTRIBUTIONS

Definition of *General-Purpose* Telepresence The past systems described in Section 1 and others can be generalized (in the visual sense) as complete local and remote environments that are virtually adjacent or superimposed, with precise control over which of the two is seen from any viewpoint. To be fully general, we assume that the environments are three-dimensional and that the spatial relationships between them may change continuously. We note that this definition may be broader than what is considered telepresence (it includes, for example, seeing only a small virtual inanimate object) but the distinction is not a technical one. We also note that completely virtual, untracked, and two-dimensional telepresence systems trivially fall under this definition.

Requirements for *General-Purpose* Telepresence To build a system that can reproduce any *visual* telepresence scenario under this definition, there are two basic requirements:

1. A live 3D description of the entire local and remote spaces
2. The ability to see local or remote imagery at any arbitrary position, with per-pixel control over which of the two is displayed

Proposed Implementation To meet the first requirement, we require a real-time 3D scanning system in each environment. We assume the remote environment is real, as producing a virtual environment is simpler. As in a previous approach [8, 9], we perform

scanning with the merged data of an array of commodity depth sensors; to our knowledge, this is the only method that has demonstrated real-time 3D scanning of all surfaces simultaneously at the scale of a small room.

To meet the second requirement, we follow the analysis of Kiyokawa et al. [5] and propose the use of a head-worn see-through display, as it is the only practical technology that will allow arbitrary virtual imagery to be displayed in any location in an arbitrary environment. Beyond the fact that *optical* see-though devices provide a "naturally and clearly visible" view of the local environment [5], they can also preserve eye contact (see Figure 3D), which is an important interaction for telepresence. We propose the use of conventional optical see-though devices rather than inherently occlusion-capable devices [5], as the former are much less bulky and are available as inexpensive commodity products. To preserve mutual occlusion, we look to Occlusion Shadows [1], a technique which uses projector-based lighting control to illuminate all local surfaces except those occluded by a virtual object. The method must be adapted from an optical see-through tabletop display with static local geometry to optical see-through glasses and dynamic geometry. A more recent work using this technique [10] exhibited dynamic local geometry, but capture was limited to objects' visual hulls and a static background was assumed; in this work, we demonstrate full geometric capture of an arbitrary environment.

We also note some disadvantages in the optical see-through approach: the burden of wearing a display, lower contrast in virtual imagery, and latency between directly viewed real objects and augmented virtual ones.

Contributions In the rest of this paper, we describe a prototype telepresence system built following the guidelines above, while offering the following contributions:

1. A design for a *general-purpose* 3D telepresence system (in the visual sense) with commodity hardware
2. Adaptation of projector-based lighting control to tracked optical see-through head-worn displays and dynamic geometry
3. *Fine-grained* immersion control between local and remote environments

The prototype system, of course, will not be truly general-purpose under the strict definition above (due to limited sensor, projector, tracking, and display coverage), but will be demonstrated to support a wide variety of telepresence scenarios.

3 System Overview

Physical Layout and Hardware Overview Figure 2 shows the layout of our prototype system. The two small rooms (shown in Figure 3A-C) are physically separated, but appear virtually overlaid as shown in the diagram. Two Microsoft Kinect depth sensors used for 3D scanning and two projectors used for lighting control are installed in room 1, while seven Kinects (and no projectors) are installed in room 2. This configuration allows demonstration of the full capabilities of our system (full environment scanning of the remote room and dynamic lighting control) to the viewer in room 1, while offering a limited experience (scanning of only the remote user and no lighting control) to the viewer in room 2 – allowing eye contact without two full sets of equipment. In either room, Epson Moverio commercial off-the-shelf optical see-through glasses or a custom wide field of view optical see-through display[1] (illustrated in Figure 3E) can be used. The custom display consists of two smartphones and two large curved half-silvered lenses mounted to a headband frame; each eye can see approximately a 1000×450 pixel region of the corresponding smartphone display over a 57° horizontal field of view. Both rooms have a NaturalPoint Optitrack tracking system that tracks markers on the head-worn displays.

[1]Designed by Tracy McSheery of PhaseSpace and Mark Bolas of USC's MxR Lab

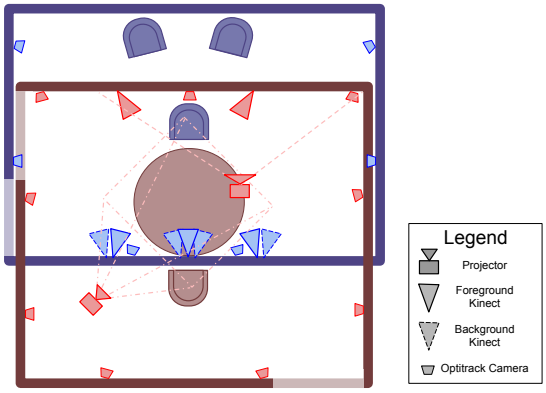

Figure 2: Layout used in our prototype system, reflecting the *virtual* arrangement of room 1 (red) and room 2 (blue).

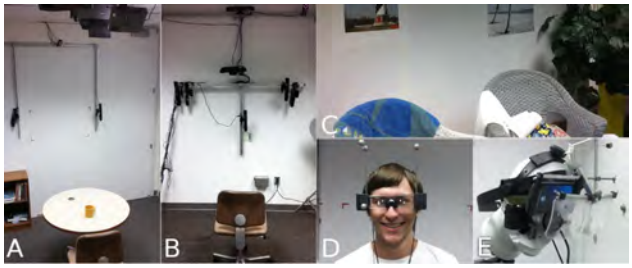

Figure 3: Photographs of equipment. A) Foreground of room 1. B) Foreground of room 2. C) Background of room 2. D) Epson see-through display. E) Custom wide field of view see-through display.

All equipment in both rooms is driven by a single PC with dual NVIDIA GTX 580 GPUs. The custom see-through device is connected to the PC through a video stream over WiFi. Our prototype currently supports only a single monoscopic user in each room as our lighting control system is designed for a single view. Plans to remove this limitation are discussed in Section 4.3.

Software Overview The system runs on 64-bit Windows 7, and OpenNI is used to communicate with the Kinect sensors. OpenGL and the OpenGL Shading Language (GLSL) are used for all rendering and GPU-accelerated processing tasks. OpenCV is used for 2D image processing functions and camera and projector calibration. NaturalPoint Optitrack Tracking Tools software was used to calibrate the tracking system and stream the positions and orientations of the tracked head-worn displays. Matlab is used to create an interpolated undistortion look-up table for the custom see-through display from measured correspondences.

4 Implementation

4.1 Calibration

System components were calibrated to an NaturalPoint Optitrack commercial tracking system installed in each room using the standard techniques described below. The coordinate system of room 2 (see Figure 2) was translated along the floor to place it in the desired position with respect to room 1.

Depth Sensors The depth sensor array in each room was calibrated and corrected for radial distortion and depth biases using previously described methods [8], except that extrinsic calibration between cameras was performed simultaneously and bundle adjustment was also performed. Each sensor array was then transformed into the corresponding tracking coordinate system by computing the transform between a common set of 3D points (corners of a checkerboard) seen by both tracking and depth capture systems.

The 3D positions of the checkerboard corners in the depth capture coordinate system were computed using OpenCV and a probe was used to determine their locations in the tracking coordinate system.

Head-Worn Displays The custom head-worn display suffered from significant geometric distortion that did not fit a standard radial model, so a procedure was developed to correct arbitrary smooth distortion. A checkerboard pattern was shown through the display and an image of the distorted pattern was recorded with a camera placed at the eye position (see Figure 3E). The corners of the deformed checkerboard were detected using OpenCV and a few extra corners around the view periphery were identified manually. The correspondences between the ideal (checkerboard) coordinates and their observed positions in the display were used to compute a cubically interpolated per-pixel look up table mapping image coordinates to camera coordinates. The Epson glasses did not have any noticeable distortion and thus no correction was performed.

To calibrate each head-worn display to the tracking system, a small marker was moved throughout the room while at each position the user moved a cursor until it aligned with the marker as seen through the head-worn display. The correspondences between the 3D positions of the marker (seen by the tracking system) and their 2D projections (cursor positions) were fit to a combined extrinsics and projection matrix mapping 3D points from the tracked head-worn display coordinate system to their 2D projected positions.

Projectors Projector intrinsics were measured by manually moving the corners of a projected checkerboard image to match the physical corners of a checkerboard placed in multiple positions. After aligning each checkerboard, the image sent to the projector was saved, and the set of images were used as input to the OpenCV camera calibration routine.

At one of the checkerboard positions, the 3D positions of the corners in projector space were computed by OpenCV and the positions were subsequently measured in tracker space using a positional probe. These point correspondences were fit to a transform from projector to tracker space.

For projectors facing a wall in which no real-time depth measurements are available (e.g. the top most projector in Figure 2), we assumed the projection surface was planar and measured the 3D positions of the corners of the projected image.

4.2 3D Reconstruction

Each room shown in Figures 2 and 3 is instrumented with multiple Kinect color-plus-depth sensors to build a real-time 3D description of the environment. These descriptions serve two purposes: to allow users to see the remote scene from their own perspectives and to allow local physical objects to occlude remote virtual ones.

To create a unified model of each room, we use a previous approach [8] to smooth and patch holes in the depth maps, create triangle meshes, and blend and color-correct data from separate sensors. The room models are combined into a shared coordinate space (see Section 4.1 and Figure 4F) and rendered from each tracked user's point of view. For each rendering, local scanned geometry is drawn black (i.e. transparent in the optical see-through displays) while virtual imagery is drawn using normal color textures. This causes virtual geometry to be erased when it is occluded by local geometry when the geometries are combined with hidden surface removal.

Following previous work [9], if parts of the environment are known to be static, live update of one or more sensors can be disabled to increase performance and reduce temporal noise and multi-Kinect interference.

4.3 Lighting Control

A limitation of most optical see-through displays is that overlaid images appear transparent unless viewed against a dark background, preventing opaque virtual imagery in a lit room. Occlusion-capable see-through displays exist [5], but are currently bulky.

Inspired by the precise light control afforded by projectors to texture surfaces in spatially augmented reality [11] and by the use of projector-based lighting to resolve occlusion for optical see-through tabletop displays [1], we propose an alternative approach in which projectors are used to selectively illuminate physical objects throughout a room. Like Bimber and Fröhlich [1], we start with a darkened environment and use an array of projectors to illuminate all surfaces that are not occluded by a virtual object with respect to the tracked user. However, we apply this technique to a different mode of operation (a viewer looking out through a head-worn display, rather than into a tabletop display), and adapt it for use with a *dynamic* local geometry through the following simplified shader-based implementation:

1. Render the scene from the perspective of the projector using the local sensor data and save Z-buffer as a depth map.
2. Render the scene from the perspective of the viewer using the remote sensor data and save Z-buffer as a depth map (generated as part of the 3D reconstruction process).
3. For each pixel in the projector image, project the corresponding depth value from the projector depth map from step 1 onto the viewer depth map from step 2.
4. If the viewer depth value from step 3 represents a closer depth value than the corresponding projector depth value, draw the pixel as black, otherwise as white.

When the projector mask is complete, we also fill any small holes (i.e. missing depth values) and apply a small blur. This step reduces two distracting artifacts: bright light shining through non-existent holes in virtual objects, and hard projector mask edges that are visible due to small calibration or tracking errors. Figure 4E shows a silhouette to be displayed behind the virtual image of a remote user.

Extending to Multiple Views As noted in Section 3, this approach is limited to a single monoscopic user in our proof-of-concept system. Multiple views can be achieved through time-multiplexing with high speed projectors and synchronized viewer-worn shutters [7]. Since our application uses only white projected light, we could achieve high performance with common color-sequential projectors (e.g. DLP projectors) by removing the color filters – obtaining an increase in brightness and framerate that would offset the corresponding losses from time-multiplexing. An additional time slice with all shutters open, projectors turned off, and the eyes illuminated could be used to preserve eye contact.

4.4 Immersion Control

As noted in Section 2, a *general-purpose* telepresence system must allow per-pixel control between real and virtual environments. Our hardware configuration allows such control; however, it is necessary to provide some intuitive software control over which environment is displayed to create the desired telepresence scenario.

Past systems have demonstrated *switchable* immersion modes; for example, SeamlessDesign [6] demonstrated switchable augmented reality and virtual reality modes for a collaborative workspace application. We extend this concept to *continuous* control between local and remote environments. In our test system, this is implemented using simple depth-based segmentations that allow continuous control over where the real environment ends and the virtual one begins. As shown in Figure 4A-D, even this simple method allows the reproduction of key telepresence scenarios: the remote user appearing *inside* the local environment, the remote user and his environment appearing to *extend* from the local environment, and an *immersive* view into the remote environment. In the future, we plan to extend this method to more sophisticated and automatic *object-aware* and *context-aware* segmentations.

5 Results

In the following section, we discuss the results obtained with our prototype system. All listed results were obtained with the custom head-worn display described in Section 3. All images were taken with a camera placed behind the display and were cropped approximately to the active display area (or smaller). Similar results were

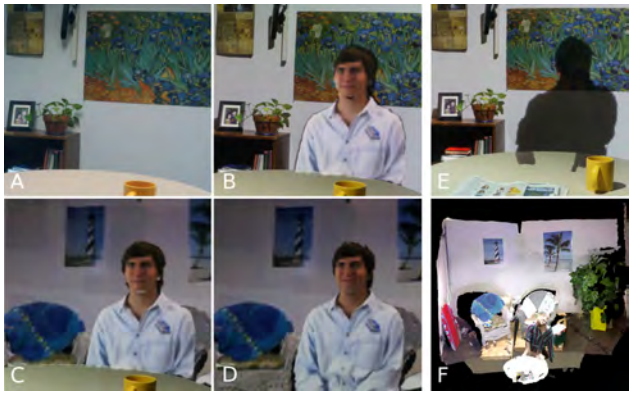

Figure 4: A-D) Configurable levels of immersion across the VR continuum from reality to virtuality displayed on our prototype system through an optical see-through display. E) Lighting control creates a dark mask to be displayed behind the virtual image of remote user. F) Combined 3D scene of the local table and remote environment.

obtained with the off-the-shelf Epson glasses, except that their field of view is much narrower.

Tracking and Calibration Figure 1E shows a user sitting next to the configured position of his virtual representation, showing that virtual imagery is registered to the room and appears with the correct perspective and scale. Figures 1B-C and 4B-C show that virtual imagery is occluded by a local object (the table), and that both the room and virtual imagery appear opaque – indicating proper lighting control. However, small misalignments in the occlusion mask (virtual imagery extends slightly over the table) and projector lighting mask (thin silhouette around virtual imagery) are visible.

See-Through Display Figure 1B-C and Figure 4B show a combination of real objects (table, mug, rear bookcase and wall) and a virtual object (remote participant). The virtual imagery appears opaque and mutual occlusion is demonstrated (the local table occludes the remote participant, and he occludes the rear bookcase and poster). Figure 1E shows that the contrast of virtual imagery is acceptable, but noticeably lower than local objects. Optical geometric distortion was not evident after correction.

3D Reconstruction Figure 1 and Figure 4B-D show the 3D reconstruction quality of the remote scene obtained with our prototype system. Most of the holes caused by multi-Kinect interference are filled, and the data from the seven utilized depth sensors is blended smoothly. However, we observe a roughness of object edges and some occlusion holes (e.g. under user's chin). We have also observed moderate temporal noise in the background data if live update of the corresponding sensors is enabled. It is also evident that the local and remote lighting do not match.

Performance Latency in our prototype system is high – due much in part to the compressed wireless video link between our PC and the smartphone display used in the custom head-worn display. This latency causes a "swimming" effect and temporary misalignment of the virtual imagery and projector lighting mask when the tracked viewpoint moves quickly. These effects could be mitigated with a direct-drive display and predictive tracking.

In a minimal configuration with three outputs (one projector and two head-worn displays) and five depth sensors (two in room 1 and three in room 2) the average display rate was 18.7 Hz. In the current maximal configuration with four outputs (two projectors and two head-worn displays) and nine depth sensors (two in room 1 and seven in room 2), the average framerate was 10.4 Hz. Please see the supplemental video for a qualitative performance measure.

6 Conclusions and Future Work

In this paper we described the design of a *general-purpose* telepresence system that is applicable to wide variety of scenarios. A prototype system utilizing head-worn optical see-through displays allows users to see remote participants and their surroundings registered to the local environment from the correct tracked perspective and with mutual occlusion. Projector-based lighting control allows remote objects to appear opaque in a lit room. Fine-grained immersion control allows the system to demonstrate several key telepresence scenarios ranging from augmented reality to total immersion.

Although our proof-of-concept system demonstrates the desired basic functionality, several components need improvements. The user experience would benefit from optical see-through displays that combine the compact size of the Epson glasses with the wide field of view of the custom head-worn display. Latency could be reduced by using a direct connection to the head-worn displays along with a Kalman filter for predictive tracking. Time-multiplexed lighting control could allow stereo views and multiple users in each room.

We are excited at the prospect of future optical see-through systems with the display form factor and function of ordinary eyeglasses that enable users to communicate seamlessly among a variety of telepresence modes throughout their daily activities.

Acknowledgements

The authors wish to thank Tracy McSheery of PhaseSpace and Mark Bolas of USC's MxR Lab for providing the custom HMD prototype, which was developed under STTR contract from ONR. We also thank Kurtis Keller and John Thomas for engineering advice and support. This work was supported in part by the US National Science Foundation (award CNS-0751187), the BeingThere Centre (a collaboration of UNC Chapel Hill, ETH Zurich, NTU Singapore, and the Media Development Authority of Singapore), the China Scholarship Council, and the Natural Science Foundation of China (No. 60970051, 61173105).

References

[1] O. Bimber and B. Fröhlich. Occlusion shadows: using projected light to generate realistic occlusion effects for view-dependent optical see-through displays. In *ISMAR*, 2002.

[2] S. J. Gibbs, C. Arapis, and C. J. Breiteneder. Teleport towards immersive copresence. *Multimedia Systems*, 7:214–221, 1999.

[3] M. Gross, S. Würmlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. Van Gool, S. Lang, K. Strehlke, A. V. Moere, and O. Staadt. blue-c: a spatially immersive display and 3d video portal for telepresence. *ACM Trans. Graph.*, July 2003.

[4] A. Jones, M. Lang, G. Fyffe, X. Yu, J. Busch, I. McDowall, M. Bolas, and P. Debevec. Achieving eye contact in a one-to-many 3d video teleconferencing system. *ACM Trans. Graph.*, 28:64:1–64:8, July 2009.

[5] K. Kiyokawa, M. Billinghurst, B. Campbell, and E. Woods. An occlusion-capable optical see-through head mount display for supporting co-located collaboration. In *ISMAR*, 2003.

[6] K. Kiyokawa, H. Takemura, and N. Yokoya. Seamlessdesign for 3d object creation. *IEEE MultiMedia*, 7(1):22–33, Jan. 2000.

[7] A. Kulik, A. Kunert, S. Beck, R. Reichel, R. Blach, A. Zink, and B. Froehlich. C1x6: a stereoscopic six-user display for co-located collaboration in shared virtual environments. In *SIGGRAPH Asia*, 2011.

[8] A. Maimone, J. Bidwell, K. Peng, and H. Fuchs. Enhanced personal autostereoscopic telepresence system using commodity depth cameras. *Computers & Graphics*, 36(7):791 – 807, 2012.

[9] A. Maimone and H. Fuchs. A first look at a telepresence system with room-sized real-time 3d capture and large tracked display. In *ICAT*, nov 2011.

[10] K. Murase, T. Ogi, K. Saito, and T. Koyama. Correct occlusion effect in the optical see-through immersive augmented reality display system. In *ICAT 2008*.

[11] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs. The office of the future: a unified approach to image-based modeling and spatially immersive displays. In *SIGGRAPH*, 1998.