# Room-sized Informal Telepresence System

Mingsong Dou, Ying Shi, Jan-Michael Frahm, Henry Fuchs* Bill Mauchly, Mod Marathe †
University of North Carolina at Chapel Hill CISCO

**ABSTRACT**

We present a room-sized telepresence system for informal gatherings rather than conventional meetings. Unlike conventional systems which constrain participants to sit in fixed positions, our system aims to facilitate casual conversations between people in two sites. The system consists of a wall of large flat displays at each of the two sites, showing a panorama of the remote scene, constructed from a multiplicity of color and depth cameras. The main contribution of this paper is a solution that ameliorates the eye contact problem during conversation in typical scenarios while still maintaining a consistent view of the entire room for all participants. We achieve this by using two sets of cameras–a cluster of "Panorama Cameras" located at the center of the display wall and are used to capture a panoramic view of the entire room, and a set of "Personal Cameras" distributed along the display wall to capture front views of nearby participants. A robust segmentation algorithm with the assistance of depth cameras and an image synthesis algorithm work together to generate a consistent view of the entire scene. In our experience this new approach generates fewer distracting artifacts than conventional 3D reconstruction methods, while effectively correcting for eye gaze.

**Index Terms:** I.4.6 [Computing Methodologies]: Image Processing and Computer Vision—Segmentation; H.4.3 [Information Systems Application]: Communications Applications—teleconferencing

## 1 INTRODUCTION

In this paper, we are exploring how far we can go in the direction of building a robust real time room-sized informal telepresence system with current technology. We want the people shown on the display be life-size and we do not limit people's movement, so it is a system for informal gatherings while the displays might take up the whole side of a wall at a break room or a entertainment space. Most importantly, the system should render a proper view for each participant.

Due to the difficulty of acquiring a reasonable 3D structure of the scene with existing algorithms, we turn to an image synthesis method based on the images captured from multiple cameras that locate at various spots on the display wall. We only apply operations that do not change the spacial relations between image pixels, such as global warping, pixel removing, hole filling, and blending.

We use multiple 2D display panels to compose a telepresence wall. Thus the rendered image shown on the 2D displays should be proper for all the viewers. Generally, people at the back want to have a sense of the consistent room, while the front people are likely engaging in conversations and the correct eye gaze should be guaranteed for them. To achieve correct eye-contact effect without breaking the sense of a consistent room, two sets of cameras are employed in our system. The first set of cameras, which are called Panorama Cameras, locate at the center of the display wall and are used together to capture a panorama view of the room. The second set of cameras, which are called Personal Cameras, are distributed evenly on the display wall and used to capture the front view of nearby persons. An image is synthesized from these two sets of cameras by superposing the front view of foreground persons onto the panoramic background image. In this way, the imagery from panorama cameras gives users a sense of a consistent room, while the personal cameras guarantee a proper eye gaze.

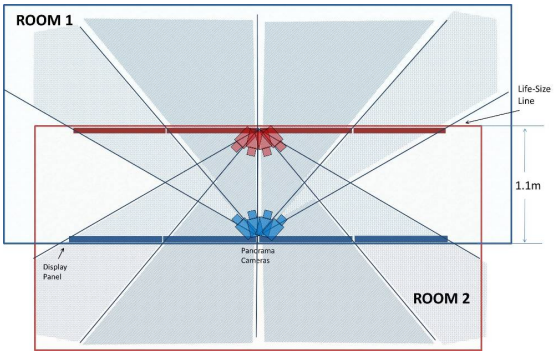


Figure 1: Place two remote rooms together with overlap.

Segmentation plays an important role in our system. Different from other segmentation tasks, we only segment out the people that stand near the display wall from the remaining scene, because these people are most likely to be communicating with the people in the other site and thus their eye gaze should be corrected. We treat the people at the back simply as background, reducing unnecessary processing. Clearly, depth information is necessary to distinguish the front people from those at the back. We use the consumer depth cameras, such as Kinect, to assist us in segmentation.

Our system works as follows. We assume that when one person wants to have a conversation with the remote people, he/she walks toward the display wall until he is reasonably close. At some point the system will perform the segmentation and switch this person's imagery from Panorama Camera to the imagery from the Personal Camera. To make the transition less noticeable, the person's segment from the personal camera is placed at the same spot where the person is in the panorama camera, and an view morphing algorithm is used to render intermediate novel views. Thus there is no jump in this person's image position during transition, and the view is gradually switched from panorama to personal.

### 1.1 Related work

Besides the commercial videoconferencing products, such as CISCO and Tandberg's telepresence system, there exists research in the academic society mainly focusing on solving eye contact problem. In [10], a carefully designed system with half mirrors was presented. Although the proper eye contact for multiple users is achieved, the system suffers from various limitations, such as difficulties of scaling up and people having to sit at specific spots. In [6][7], a spinning mirror system is used as a 3D display to achieve eye contact in the one-to-many video conference, where the remote person should sit still as well and only the facial part of the people is shown on the 3D display. In [2][8][1][9], various videoconferencing systems are built based on the concept of the

*e-mail: {doums, yshi, jfm, fuchs}@cs.unc.edu
†e-mail: {bmauchly, mmarathe}@cisco.com

IEEE Virtual Reality 2012
4-8 March, Orange County, CA, USA
978-1-4673-1246-2/12/$31.00 ©2012 IEEE

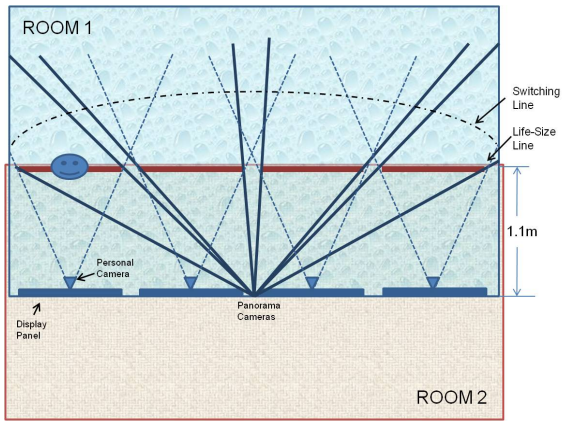


Figure 2: Top-Down view of the system. For clarity, we only show the cameras for one site.

shared virtual table. We extend this idea to build a shared virtual room allowing participants to walk around freely. There are other room-sized virtual environment systems build, such as CAVE [3] and BLUE-C [4], but they are not designed for video-conferencing.

## 2 Hardware

As shown in Fig. 5, four 65-inch 1028p monitors are placed side by side to compose a telepresence wall which is around 1.43 meters high and 3.67 meters wide (the bezels are taken into consideration), covering almost the whole side of a wall. As mentioned earlier, to show on the display the correct eye gaze without breaking the sense of a consistent room, we have placed two sets of cameras in front of the display—Panorama Cameras and Personal Cameras.

**Panorama Cameras.** Four panorama cameras locate closely at the center of the display wall. These cameras point toward different directions to cover most of the room. A panoramic image of the room can be synthesized from these cameras given camera calibration parameters. In Fig. 1 two remotely located rooms are drawn together with overlap. We point the cameras in such a way that each of them must "see" one whole display panel of the other site. Overlapping two rooms instead of placing them side by side gives us some insights of the system. The connection between display panels and Panorama cameras are very clear in the drawing, i.e., the image of one camera is only shown on its corresponding panel at other site.

In the software, to show the panoramic view of the room on the display wall, we turn each camera into a projector with the same location and orientation and project the capture images onto the corresponding virtual display panel. Thus, the people that stands at the line where the display panels locate in the drawing are shown in life-size on the display wall. This line is called life-size line, and its location is an important parameter determining some other system configurations, such as the field of view of the panorama cameras. We set the life-size line 1.1 meters away from the panorama cameras due to the fact that 1.1 meters is a comfortable distance for people engaging in a conversation and we want them to be shown in life-size on the display wall. The chosen life-size line gives us an overall angle of view of $118^\circ$ for the panorama cameras.

**Personal Cameras.** We expect people engaging in a conversation would stand right in front of each panel instead of the gaps between panels. Hence, one Personal Camera is placed at the center of each display panel to capture the front view of the people as shown in Fig. 2. Each Personal Camera has the angle of view of $47^\circ$ and covers the area in front of its corresponding panel. All the cameras are placed at the average human eye-height vertically, around 1.71 meters high, and are tilted downside by 14.7 degrees.

Note that in the above setup, each panel at one site is associated with two cameras in the other site—one Panorama Camera and one

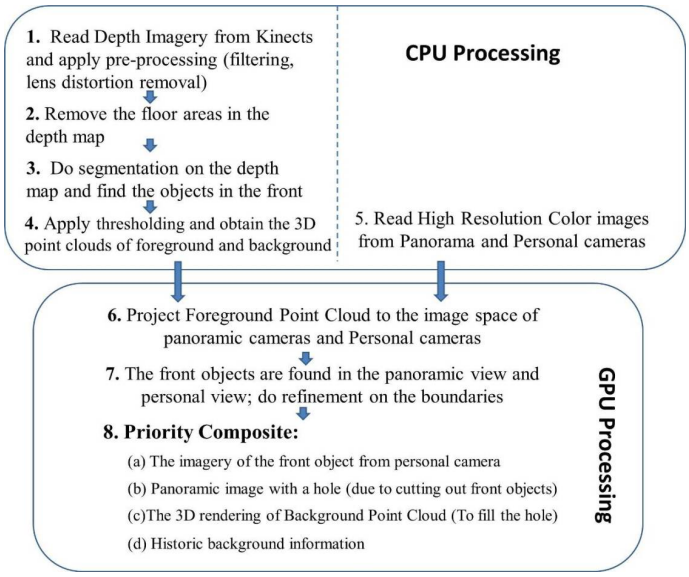


Figure 3: Camera Switching Algorithm for one display panel.

Personal Camera. The images from these two cameras are only shown on its associated panel, and the panel only shows images from these two cameras. This configuration simplifies the algorithm due to its parallel structure, and also makes it possible to scale up the system easily.

**Kinect sensors.** Robust and real-time segmentation is important to solve the eye-contact problem in our algorithm. Besides, we need the depth information to differentiate the people in front from the people at the back. We use Kinects to assist us in segmentation. Kinect is a consumer depth camera which outputs a dense depth map 30 fps with the resolution as high as $640\times480$. Totally six Kinects are used at each site to cover the room.

All the cameras, including color cameras and depth sensors, are calibrated under the same world coordinate system [11]. The position of each display panel is also measured under the same world space.

## 3 Camera Switching System

The heart of our system is the camera switching algorithm. We choose the suitable cameras for people locating at various places, thus the proper eye contact could be guaranteed for the people standing near by the display wall while a consistent background is maintained. As mentioned earlier, we achieve this via image segmentation.

In an uninteresting case when nobody stands in front, the system simply shows a panoramic view of the room by stitching images captured by panorama cameras. When a person walks toward the display wall to have a conversation with others in the other site and he is closer than some distance to the display wall (the switching line is shown in Fig. 2), the personal camera right in his front is turned on. The system then treats this person as a foreground object and performs segmentation on both the panorama camera and personal camera. The person's imagery is first cut out from the panoramic image and is replaced by the image segment from the personal camera. The image segments of this person in the panorama camera and personal camera generally have different shapes as shown in Fig. 4, so after the above replacement operation there are some holes in the synthesized image. With the assistance from the depth sensors, part of missing pixels are recovered by image rendering from the captured 3D structure. For the pixels with no depth measurement, the historic background information is used to fill the holes.


Figure 4: Various Intermediate results for one display panel during the operation of the system. (a) the image from one panorama camera after correcting lens distortion; (b) the image from one personal camera; (c) the foreground point cloud and background point cloud from depth sensors after performing segmentation; (d) the segmented foreground in the personal camera; (e) the segmented background in the panorama camera; (f) merge (d) and (e); (g) the synthesized image after hole filling.

**Algorithm Details.** As mentioned before, we could treat each panel separately and run exactly the same algorithm on them because each panel is independently associated with one panoramic camera and one personal camera. Fig. 3 shows various steps of the algorithm. There are two processing blocks—CPU Processing and GPU Processing.

CPU Processing includes data acquisition and other operations that are hard to be parallelized. The captured color images are directly sent to GPU without further processing. As to the depth maps from Kinects, a serial of operations are applied to get the foreground point cloud and background point cloud, which are then sent to GPU. These operations include depth map filtering, lens distortion removal, segmentation, performing thresholding to get foreground segments, converting depth map to point cloud, and etc. Filtering includes performing morphology operations (close operation) to fill some holes in the depth map and median filter to reduce the noise. Thresholding is to pick the foreground object, whose centroid is closer than the switching line to the display wall. Segmentation on the depth map will be discussed later.

GPU Processing takes color images, foreground point cloud and background point cloud as inputs. First, lens distortion is removed for the color images, and then segmentation on them is achieved by projecting the foreground point cloud to the image space of panorama and personal cameras. Next, some refinements are performed to obtain better foreground boundaries. Finally, several image layers are composited to one image which will be projected onto the virtual displays from the panorama camera's perspective. When composing images, alpha channel blending is used to smooth out jagged boundaries in the segmentation. Various intermediate results for one display panel of the system is illustrated in Fig. 4.

Different from other segmentation tasks, we define the foreground as the close-up objects. The depth information helps us to achieve this task. As shown in Fig. 3, we fist identify the foreground object in the depth map; then acquire the foreground point cloud; next the point cloud is projected to 2D image space of panoramic and personal cameras, and the areas covered by the point cloud after projection are treated as foreground in the color images. Finally, some refinements on the foreground is performed to align its boundaries with the edges in the color image.

**Segmentation on Depth Map.** To identify the foreground objects on the depth map, we first eliminate the floor pixels. When we perform the camera calibration, the $XOY$ plane of the world coordinate system is defined as the floor surface. Therefore all the pixels with small $z$ values in the world coordinate system are classified as floor pixels. Then we run connected component labeling algorithm on the remaining pixels. Here, two pixels are connected if they are neighboring pixels and have a similar depth value. Finally, thresholding is applied on the centroid of the each connected component. Only those closer than the switching line to the display wall are classified as foreground. Note that thresholding is applied to the whole component instead of each pixel, which prevents splitting a person into foreground pixels and background pixels when he/she stands near the switching line.

**Segmentation on Color Images** The above segmentation result is in depth sensor's perspective, and must be converted to the result in the perspective of color cameras. Given camera calibration parameters, the foreground segments in the depth map are first converted to 3D point clouds, which are then back-projected to the image spaces of the color cameras. The 3D rendering pipeline implemented in GPU is used to fulfill this task. We use GL shader language to render these points from the color camera's perspective. The points are resized so that neighboring points touch each other in the rendered image without leaving gaps between points [5].

The foreground mask resulted from above GPU rendering generally has jagged edges due to the noise from depth sensors. To make the segmentation results more appealing, we first apply morphology operations, and then blur the foreground mask so that inner pixels of a segments have values of 1.0, and pixels lying around segment boundaries have values between 0 and 1.0 depending on their distances to the boundary. All other background pixels far away from boundaries have values of 0. This foreground mask will be treated as the alpha channel of its corresponding texture image. Thus, the jagged boundaries are smoothed when applying alpha channel blending. In addition, we slightly adjust the alpha value around boundary pixels based on its color. If one pixel has similar color with inner foreground pixels, we increase its alpha value, otherwise we decrease its alpha value.

**Transition** Since the panoramic cameras and personal cameras have different positions and orientations, a sudden camera switch leads to the jump of people's position and orientation in the synthesized image. To make the switching less artificial, we render some intermediate views. Given the point cloud of foreground objects,

Figure 5: Comparison between the panoramic view and synthesized view of the camera switching algorithm. (a) the panoramic view with a side-facing person; (b) the synthesized view with the panoramic background and the front-facing person.

the novel views are generated from various viewpoints between the panorama camera and the personal camera.

## 4 Results

We constructed two display walls at University of North Carolina at Chapel Hill. We ignored the network transmission by connecting the display panels from one site and its remote cameras at the other site to the same machine (two display walls locate at adjacent rooms). Specifically, we have four machines, and each machine is responsible for devices of half of the room—two adjacent panels, four remote high resolution cameras, and three Kinects that cover the same half of the room. This is a huge computation load for one machine, and the algorithm runs around seven frames per second. We are currently adding more machines to split the computation burden.

The main contribution of this paper is to provide an method to guarantee proper eye contact for the front people engaging in conversations without damaging the sense of the consistent room for other participants. Fig. 5(a) shows a panoramic view of the scene where one person stands at the left-most panel and looks straight ahead. Clearly, in the panorama image, the person looks aside, since the panorama cameras are not locating right in front of the person. By synthesizing an image from all the panoramic cameras and personal cameras, we have a front-facing person as shown in Fig. 5(b). More results are provided on the supplemental materials.

## 5 Conclusion and Limitations of the System

In this paper, a room-sized informal telepresence system for informal gathering is presented. By synthesizing images from two sets of cameras—Panorama Cameras and Personal Cameras, the proper eye contact during conversations is guaranteed while a consistent background is maintained.

**Remaining Problems.** First, the segmentation is not perfect, leading to noticeable halo effects. Second, although some processing is employed, the transition is still not natural enough, mainly due to the artifacts in the generated novel views. In addition, the limitation of the current system is that we did not achieve the exactly true eye contact. If the two persons at two sites both stand right in the middle of the corresponding panels and their remote agent cameras are placed at their eye-height, the true eye contact is achieved. Generally, this is not the case. However, the eye contact problem is ameliorated by adding a front-facing camera for each display panel.


## Acknowledgements

We would like to thank Jonathon Bidwell, Peter Lincoln and Andrew Nashel for useful discussions, Herman Towels and John Thomas for mechanic supports.


## References


[1] N. Atzpadin, P. Kauff, and O. Schreer. Stereo analysis by hybrid recursive matching for real-time immersive video conferencing. *IEEE Trans. on Circuits and Systems for Video Technology*, 14(3), 2004.

[2] H. H. Baker, N. Bhatti, D. Tanguay, I. Sobel, and et al. Computation and performance issues in coliseum, an immersive videoconferencing system. *ACM International Conference on Multimedia*, 2003.

[3] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti. Surround-screen projection-based virtual reality: the design and implementation of the cave. *SIGGRAPH*, 1993.

[4] M. Gross and et al. blue-c: A spatially immersive display and 3d video portal for telepresence. *ACM Trans. on Graphics*, 22(3), 2003.

[5] D. T. Guinnip, S. Lai, and R. Yang. View-dependent textured splatting for rendering live scenes. *SIGGRAPH 2004 Posters*.

[6] 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 Transactions on Graphics*, 28(3), 2009.

[7] A. Jones, I. McDowall, H. Yamada, M. Bolas, and P. Debevec. Rendering for an interative 360 light field display. *ACM Transaction on Graphics*, 26(3), 2007.

[8] P. Kauff and O. Schreer. An immersive 3d video-conferencing system using shared virtual team user environments. *International Conference on Collaborative Virtual Environments*, 2002.

[9] O. Schreer, I. Feldmann, N. Atzpadin, P. Eisert, P. Kauff, and H. Belt. 3dpresence—a system concept for multi-user and multi-party immersive 3d videoconferencing. *European Conference on Visual Media Production*, 2008.

[10] L. C. D. Silva, M. Tahara, K. Aizawa, and M. Hatori. A teleconferencing system capable of multiple person eye contact (mpec) using half mirrors and cameras placed at common points of extended lines of gaze. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 470–479, 1995.

[11] Z. Zhang. A flexible new technique for camera calibration. *TPAMI*, 22(11):1330–1334, 2000.