International Conference on Artificial Reality and Telexistence
Eurographics Symposium on Virtual Environments (2017)
R. Lindeman, G. Bruder, and D. Iwai (Editors)

Real-time Visual Representations for Mixed Reality Remote Collaboration

Lei Gao[†1], Huidong Bai[‡1], Thammathip Piumsomboon[§2], Gun A. Lee[¶2], Robert W. Lindeman[‖1], Mark Billinghurst[**2]

[1]The Human Interface Technology Laboratory New Zealand (HIT Lab NZ), University of Canterbury, New Zealand
[2]School of Information Technology and Mathematical Science, University of South Australia

Abstract

We present a prototype Mixed Reality (MR) system with a hybrid interface to support remote collaboration between a local worker and a remote expert in a large-scale work space. By combining a low-resolution 3D point-cloud of the environment surrounding the local worker with a high-resolution real-time view of small focused details, the remote expert can see a virtual copy of the local workspace with an independent viewpoint control. Meanwhile, the export can also check the current actions of the local worker through a real-time feedback view. We conducted a pilot study to evaluate the usability of our system by comparing the performance of three different interface designs (showing the real-time view in forms of 2D first-person view, a 2D third-person view and a 3D point cloud view). We found no difference in average task performance time between the three interfaces, but there was a difference in user preference.

CCS Concepts

•***Human-centered computing*** → ***Mixed / augmented reality; Virtual reality; Collaborative and social computing design and evaluation methods;***

1. Introduction

This paper describes a system for capturing and sharing a user's local environment with a remote collaborator. With the growth of global high speed networks, real-time remote collaboration tools enable people to work together across large distances. A typical system connects a local workspace where a worker needs help with an unfamiliar physical task to a remote expert who can provide guidance via cues such as speech, pointing or virtual annotation. For example, the Remote AR software [KLSB14] allows a local worker to share video of their workspace with a remote expert who can place annotations on it. In systems like this, 2D video is often used to show the remote expert about the local worker's current situation. However, without a true three dimensional view of the local scene, the remote expert may not be able to correctly understand the spatial relationships of the local environment.

Videos from a fixed position [KRF07] [AL11] can provide an overview of the entire local workspace. However, with the limited camera view range and angle, this kind of remote collaboration systems often requires both the local worker and remote expert to focus on simple tasks in a small area. But in some real-world scenarios, the local worker may need to handle different devices located around a large work environment, such as several machines in a factory room.

In another example, a head-mounted camera could show the local worker's view to the remote expert as he/she moves through the workspace [FSK03] [KLSB14] [GLB16], but the shared scene will always be limited to what the local worker sees, and does not support independent viewpoint switching for the remote expert. In this case, it is quite difficult for the remote expert to understand the spatial relationships between local physical objects or find objects of interest. The major motivation for this research is exploring how to help a remote expert understand the local worker's surrounding environment and enhance remote collaboration in a room-scale workspace.

To enhance remote collaboration, we developed a prototype system that reconstructs the local physical environment and shares it as a 3D Virtual Reality (VR) environment around the remote expert. Using a VR head mounted display (HMD), the remote expert can then freely explore the VR environment to better understand the spatial relationship between objects in the local workspace. In

[†] lei.gao@pg.canterbury.ac.nz
[‡] huidong.bai@canterbury.ac.nz
[§] thammathip.piumsomboon@unisa.edu.au
[¶] gun.lee@unisa.edu.au
[‖] gogo@hitlabnz.org
[**] mark.billinghurst@unisa.edu.au

this case, the remote experts may feel as though they are sharing the same workspace as the local workers. Based on the HMD position tracking, the expert can navigate himself/herself through the virtual copy of the local worker's real environment.

Our remote collaboration system captures and reconstructs the local scene as a static 3D point cloud set. Once created, there is no real-time update of the point cloud from the local worker's side. However we developed three different interface ideas to provide real-time feedback to show the local worker's actions. We also conducted a pilot study to compare these three different interfaces. Overall, our research has the following novel aspects:

- A novel remote collaboration system that combines AR, VR and 3D space capture,
- Independent viewpoint control for remote collaboration in a room-scale virtual work environment, and
- Real-time feedback from the local to the remote location for the remote expert to check the current situation.

2. Related Work

Most of the current research in the field of remote collaboration tends to focus on reproducing a face-to-face collaborative experience [GXS*12]. However, methods for sharing a local worker's work space and to support remote task space collaboration is becoming a major research area. To make users feel more connected to each other during remote collaborative tasks, some researchers have explored how to provide richer local context to remote experts.

Video streaming systems have been shown to be better than audio-only systems while completing collaborative tasks [FKS00]. For example, when the task involves manipulation of objects that are difficult for the users to describe verbally, a video view of the shared workspace is more valuable than audio-only communication. Based on this, researchers have explored different ways of using remote cameras for collaboration.

One typical video-sharing system uses overhead fixed video cameras on each side to combine the views of both the local and remote workspaces together, and displays the fused view on monitors in front of the users. Alem et al.'s work [AL11] has shown this setup to be effective for monitoring the progress of tasks in a constrained workspace that do not require complicated manipulation. However, this kind of system only provides a general overview of the local workspace from a fixed view direction, but little detail [FKS00].

In order to overcome the limitations of fixed-view systems, researchers have tried to provide a dynamic view of the local environment via head-mounted cameras, hand-held cameras or switching between multiple cameras. Previous studies suggest that automatically following the local worker's actions can effectively reduce the remote expert's cognitive load [LSCG13]. In this case, some researchers [FSK03] [KLSB14] started using head-mounted cameras on the local side as a video capturing device, enabling the remote expert to keep focus on the worker's first person view. Other researchers [RBB07] tried to automatically track an indicator, such as the worker's hand, to provide a useful view of where the local actions mostly take place.

Using a head-mounted camera to capture and share the local worker's first person view restricts the remote expert to seeing only what the local worker sees. Frequent movement of the local worker's head may also create rapid changes of the point of view (POV) [LSCG13]. In other words, the view captured by a head-mounted camera can be quite unstable and highly distracting for the remote expert. In some cases, a static wide-angle camera can provide better performance than a head mounted camera during remote collaborative tasks [FSK03].

To overcome this limitation, some of the researchers have begun to explore how 360° video [SFG*16] [KNR14] and panorama imagery [BNR14] [SKY*12] can be shared to allow the remote expert to have an independent view into the remote space. Other researchers have focused on how depth sensors could be used to create 3D models of the local worker's workspace, enabling the remote expert to view the space in 3D and have a completely independent point of view.

The RemoteFusion system [AAT13], was one of the first remote guidance systems to support an independent 3D viewpoint for the remote expert. Multiple depth sensors were used to capture the local work environment, which was then rendered as a 3D model in the VR world and streamed to the remote side. On the remote side, the expert could view the 3D virtual scene on a multi-touch screen and control his/her viewpoint by simply rotating the scene with two-finger touch and zoom in with a pinch gesture. A similar system from Sodhi et al. [SJF*13], called BeThere, also allowed the local worker to reconstruct their local scene by using a hand-held Kinect sensor.

While some researchers have focused on rebuilding the local scene in the VR world to provide view independence, others have tried to support remote guidance in a large workspace by using a simultaneous localization and mapping (SLAM) system. A SLAM system [TL08] can create a map of an unknown environment while simultaneously keeping track of the camera's location within it. In this case, the remote collaboration system can operate in environments of arbitrary geometric complexity, and support world-stabilized annotation and local virtual camera movements independently from the remote expert's viewpoint.

In the system presented by Gauglitz et al. [GNTH14b] [GNTH14a], SLAM tracking was used to locate the position of the local worker's viewpoint and build a world-stabilized coordinate system in the 3D space. This enabled the system to project the local worker's current view onto the reconstructed virtual model at the remote side and integrate visual symbols at 3D position within the local workspace as augmented virtual clues. Both the local worker and remote expert's viewpoints could be tracked in the same shared world coordinate system, so they could navigate their own views independently from each other.

Using a desktop computer with the above systems, the remote experts could only view the local scene through a 2D display, which still reduces the spatially connection between the remote expert and the local worker. To deal with the issue, head-mounted displays (HMD) for the remote experts may be an alternative display choice. Johnson et al.'s study [JGM15] showed that using an HMD has an advantage for remote experts, by supporting more frequent directing commands and more proactive assistance during dynamic

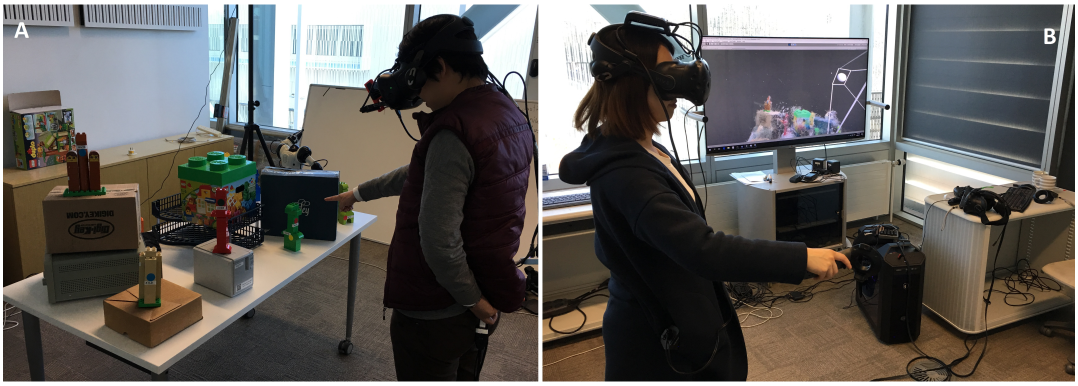

Figure 1: *Static local environment capturing with real-time feedback for Mixed Reality remote collaboration. A: the local user stands in front of a workspace, identifying a particular LEGO model which the remote helper has pointed at. B: The remote expert observes the local environment and guides local worker via pointing in the VR world.*

tasks. Results from other studies [TAH12] [ABM15] showed that remote experts feel they are spatially sharing the same work environment with the local worker while viewing the local scene through a HMD.

Previous research in the field of remote collaboration was limited by either a 2D representation of the local work environment or a small worksapce. In our research, we tried to fill these gaps in one system by capturing and rendering the entire local scene as one static 3D point cloud, and viewing it in a VR headset, so the remote expert could feel as if he/she was sharing the same virtual work environment with the local worker.

3. System Overview

In our research, our goal was to enable the remote expert to navigate in the shared virtual environment independently from the local worker's current point of view while working in a large workspace. To achieve this we use 3D space capturing and reconstruction to create a copy of the local worker's real workspace. Position tracking of the HMD increases mutual awareness between users, and visual cues and voice contact are also enabled to support natural communication, reproducing the face-to-face work experience.

Our prototype is subdivided into two logical sub-systems: (1) the local worker space in which a local worker is dealing with some unfamiliar physical tasks, and (2) the remote expert space where a remote expert provides guidance for the local worker to accomplish the task goals (Figure 1).

3.1. Local Workspace

In the local workspace, the local user wears a VR HMD (HTC Vive) for the duration of the task. A wide-angle video camera is attached to the front face of the headset and the video stream is passed through the display to create a video-see through Augmented Reality (AR) display. The local worker can directly see the surrounding scene and freely move in the physical environment while wearing the VR headset. Figure 2 shows one example of the local worker's view and the physical layout of the sensors on the HMD.

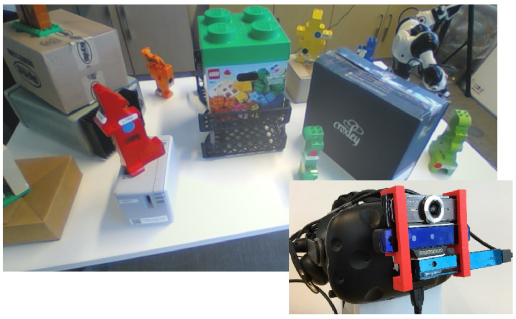

Figure 2: *Local worker's view and VR headset*

The local physical workspace is captured and rendered as a single static point cloud set. In order to achieve this, we also attached one long-range depth sensor (Intel RealSense R200 with an operating range from 0.5m to 3.5m) to the front face of the headset. Before the task starts, the local worker needs to wear the HMD and walk around the workspace to enable the reconstruction system scanning, capturing of all the details of the local physical environment. This captured data is then fused together as one dense point cloud set and streamed to the remote side. In this way, the remote expert has an overview of the entire local workspace even before the task actually starts. This process is described in more detail in section 3.3.

One issue is that the captured 3D point cloud of the local workspace is static, which means that it cannot be updated when an item's position has been changed during the task. To deal with this, we attached another short-range depth sensor (Intel RealSense SR300 with an operating range from 0.3m to 2m) to the front face of the HMD. Compared to the R200, this short-range sensor can support higher resolution depth frames and detect the distance to items even when the sensor is quite close to them. By using this sensor, we can enable real-time single frame streaming from the local to remote user, providing an update to the static 3D point cloud view.

Furthermore, using the Vive lighthouse tracking on the HMD, we can collect the local worker's head position and orientation information in real-time. By streaming this information to the remote side, the local worker's view frustum can be rendered in the remote expert's VR space to support mutual awareness for the remote expert.

The depth sensors, video camera and VR headset are directly connected via USB to a single local PC, which is responsible for local data processing, such as 3D point cloud fusing of the local scene, real-time view feedback capturing and data streaming to remote side. The Local PC is set up with an Intel Core i5, 8GB RAM and NVIDIA GeForce GTX 770 GPU, running Windows 10 system.

3.2. Remote Workspace

One user performs as the expert, guiding the other user as a novice worker in performing specific tasks within the workspace. The remote expert is also asked to wear a VR HMD (HTC Vive). The 3D point cloud of the local worker's space is wirelessly streamed to the remote side and rendered as one static virtual model in the VR environment. This allows the remote expert to have an overview of the local work environment, and to freely move around in the VR world to observe the virtual replica of the local scene from any direction. In this way, the remote expert can quickly identify any target objects while guiding the local worker. Figure 3 shows an example of what the remote expert's view looks like.

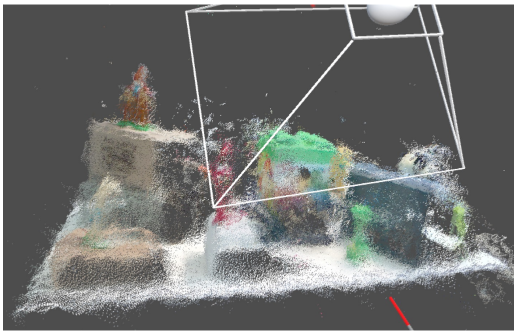

Figure 3: *Remote expert's view*

The static 3D point cloud capturing of the local scene provides the remote expert with an independent view while the local worker is working in a large (e.g. room-scale) workspace. At the same time, our system also streams the current view frame of the local worker to the remote expert to act as a reference for the remote expert to observe real-time changes in the local workspace. Most of the time, the local worker is watching what he/she is working on. Therefore, the current frame from the head-mounted camera is enough to show all the changes at one time. As described in Section 4, we designed three different interfaces for showing the real-time single frame view streamed from the worker to the expert, and then conducted a pilot study to evaluate these interfaces.

The local worker's view frustum is also shown in the remote expert's VR space (Figure 3), allowing the remote expert to see the local worker's head movement and view direction. This supports the expert in checking whether or not the worker is following the right guidance. The expert also hold one Vive handheld controller which is rendered as a wand in the VR world, and is streamed back to the local side as one virtual cue overlaid on top of the worker's view. In this way, the remote expert can provide virtual pointing feedback to help guide the local worker.

The remote headset is wirelessly connected to a PC (called "Remote PC") by using the TPCast Vive wireless adapter. This Remote PC is set up with an Intel Core i5, 8GB RAM and NVIDIA GeForce GTX 970 GPU, running Windows 10 system.

3.3. Scene Capturing

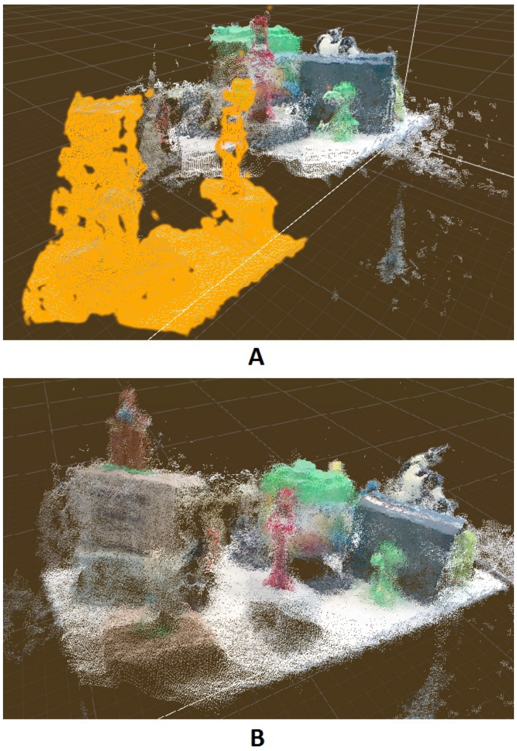

Figure 4: *Keyframe registration. A: The yellow point cloud is built based on a new frame; B: After the registration, the new frame is stitched to the previous ones*

We developed a simple method for creating the final 3D point cloud. In order to capture and render the entire local physical scene as one single dense point cloud model, we attached an Intel RealSense R200 sensor to the front face of the VR headset, facing toward the workspace. While the local worker is walking around the local workspace, the system can capture the current view based on the aligned RGB frame and depth frame from the sensor. Each pixel of this view is then projected into the Vive VR camera coordinate system by using the intrinsic parameters of the sensor, which turns the view into a dense point cloud. The position of the Vive VR camera, taken to be the same as the Vive headset, is captured by the Vive lighthouse hardware. In this case, the point cloud of each frame can be finally mapped into the Vive VR world coordinate system.

We use a keyframe based registration method to do the local scene capturing. The scene reconstruction process starts by the local worker manually pressing the trigger on the Vive controller. While the scene capturing is running, the first frame captured is considered as the initial keyframe of the system. The point cloud of each following frame is then identified as a keyframe or not based on its relevant position to the previous keyframes by using the Iterative Closest Point (ICP) algorithm [CM91]. If it is a keyframe (20% to 40% overlap with previous three keyframes), this point cloud data will be saved and stitched with the previous keyframes. While the worker walking around the local workspace, the system keeps adding new keyframes into the previous ones. After the entire local workspace is captured (as judged by the local worker), the local worker can press the trigger again on the Vive controller and the system will stop collecting new keyframes. The saved point cloud set will then be streamed to the remote expert side as one virtual replica of the local scene. Figure 4 shows an example of the keyframe registration.

The size of the local workspace that can be captured is based on the setup of the Vive Lighthouse tracking area (no more than 5m*5m). During the scene capturing, the local worker needs to move slowly in the workspace. Otherwise, keyframe registration may be failed. It takes around 30 seconds for the local worker to finish the scene capturing process in a 4m*5m size room. Since we use long range depth sensor to capture the local environment, the reconstructed 3D point cloud can only present the general geometric distribution of the local workspace in a low resolution.

3.4. Network and Rendering

The local scene is captured and rendered as a dense point cloud in the local side, and then sent to the remote side before the task starts. During the task, the system streams the local worker's headset position, along with the current view captured by the short-range depth sensor, to the Remote PC to assist the expert. At the same time , the Remote PC sends the virtual wand info as guidance to the Local PC. All of this data streaming is based on a wireless connection between the Local PC and the Remote PC. In order to achieve real-time data communication, we use the NETGEAR Nighthawk X6 WiFi Router and the sharing service supported by HoloToolkit.

The Unity game engine provides good support for VR scene rendering while using the HTC Vive headset. All of the point cloud data is rendered as vertical meshes in Unity. On the local side, the frame rate reaches 15 fps while reconstructing the local scene before the task and 30 fps during the task with real-time single frame streaming enabled. On the remote side, the frame rate reaches 45 fps during the tasks.

4. Pilot User Study Design and Method

In order to evaluate the usability of our prototype system, we conducted a pilot study comparing different real-time feedback approaches for the remote expert. The purpose of this study was to explore how different real-time visual representations of the local worker's space could improve the remote user's performance while the local worker is working in a large workspace.

4.1. Interfaces Design

In our study, we are mainly investigating the remote expert's user experience with different spatial sharing technologies. We have created a hybrid interface that combines a low-resolution 3D point-cloud with a high-resolution real-time view for small focused details. The advantage of the hybrid interface is that it provides the large-scale static 3D information at the same time as real-time 2D or 3D detail information.

Based on our current system setup, we can share two types of spatial information from the local worker to the remote expert:

1. Large surrounding background in a 3D point cloud reconstruction of the local environment before experiment tasks start.
2. Small detailed foreground with real-time feedback in 2D video or 3D point cloud based on the local worker's current view.

We created three interfaces to investigate our system. Each of them has a 3D point cloud background enabled, but with different foreground display approaches:

1. *First person view (FPV)*: The static 3D point cloud of the local scene is displayed as a background in the remote expert's VR world. Real-time 2D video of the local worker's view is displayed at the top-right corner of the remote expert's view as a 2D window, and always follows the remote expert's head movement (Figure 5:A).
2. *Third person view (TPV)*: The static 3D point cloud of the local scene is displayed as a background in the remote expert's VR world. Real-time 2D video of the local worker's view is displayed as a 2D window and attached to the local worker's head view frustum in the remote expert's VR world (Figure 5:B).
3. *Point cloud view (PCV)*: The static 3D point cloud of the local scene is displayed as a background in the remote expert's VR world. The current frame of the local worker's view is captured by the short-range depth sensor and rendered as a 3D point cloud in the VR world to show real-time feedback from the local side to the remote side (Figure 5:C). Since the point cloud of the current view is directly overlaid on top of the static point cloud set of the local scene, the remote expert could see the current changes in his/her 3D VR space directly.

4.2. Tasks Design

The participants performing as remote experts were asked to guide the local worker on finding target letters on Lego models located separately around the local workspace.

Figure 6 shows one image of the local task workspace. There were eight Lego models randomly located in the workspace along with some irrelevant objects to block the Lego models from each other. Each Lego model has one unique color (red, orange, light green, dark green, blue, yellow, white and brown), so that objects could be searched for by subjects based on the color. Each Lego model had three labels with different colors and letters on it. Like the block colors, the labels could be used for searching by their own colors too.

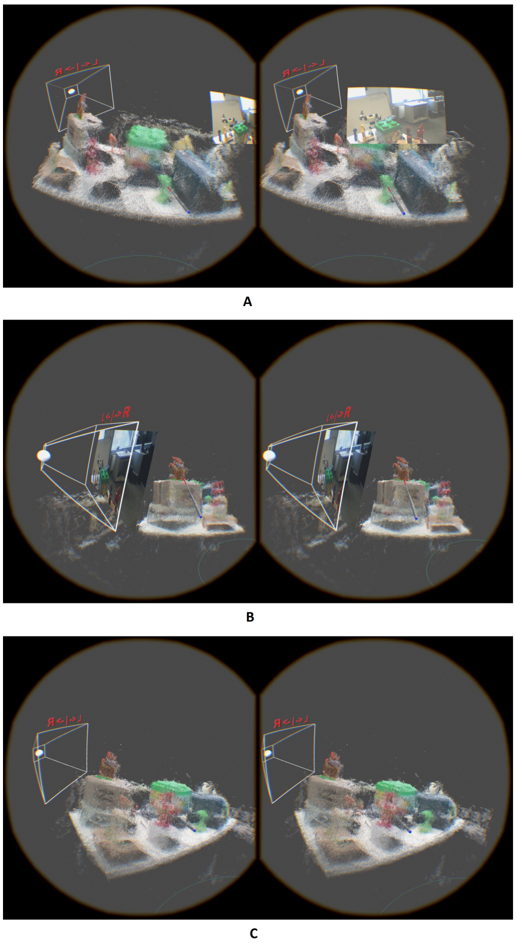

Figure 5: *Three interfaces. A: 3D point cloud background with 3D video foreground in first person view (FPV); B: 3D point cloud background with 3D video foreground in third person view (TPV); C: 3D point cloud background with 3D point cloud foreground (PCV)*

Before each task started, the system randomly picked one model color among the eight in total as the target model color, and one label color on the target model. The model color and label color were then shown on the remote expert's VR display as the object that needed to be found. Following this, the remote expert needed to guide the local worker to find the target model first based on the model color, and then locate the target label on the model based on the label color. When the local worker finally saw the right label on the right model, the remote expert needed to read the four letters on the label, and press the trigger on the controller to finish the task. The system then automatically recorded the task completion time and showed the next pair of target model/label colors to the remote expert. For each interface, the remote expert has to finish five tasks in total.

To start guiding the local worker, the remote expert first needs to find the target model himself/herself. The low-resolution 3D point-cloud background provides an overview of the entire local workspace which is a straightforward way for the remote expert to locate the model position. However, while searching for the label on the model, the resolution of the background may not be adequate. In this case, checking the real-time high-resolution foreground view would be a good choice, especially if the expert asks the worker to pick up and rotate the model for him/her to search.

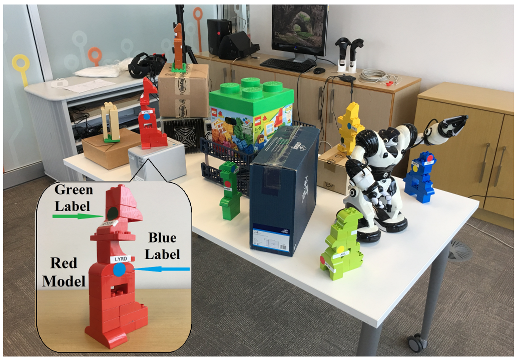

Figure 6: *Scene of the local task workspace*

During the tasks, the remote expert is required to hold one Vive controller in the hand, which is rendered as a virtual wand in the VR world. By using the wand, the remote expert can point to an objects to guide the local worker. At the same time, both the local worker and remote expert can talk to each other for communication (For the experiment, we set both the local and remote users in the same large lab room, separated by a curtain). However, the remote expert was not allowed to describe the target model color and label color directly to the local worker. The local worker can also point to physical objects by using his/her hands within the view of the head-mounted sensor, which could be seen by the remote expert through the real-time foreground view.

4.3. Participants

Since the local workers use the same video see-through interface for all three conditions, we did not measure the performance of the local worker in this user study. All participants were recruited for the role of remote expert. Ten people took part in the study, four women and six men, aged 19 to 44. Most of them had previous experience with video conferencing systems, such as Skype, Snapchat or WeChat, except one. All of the participants could identify the colors without any trouble.

4.4. Experimental Design

We used a within-subjects design with one independent variable (interface types) and one dependent variable (task completion time). At the beginning of the user study, participants were given a general explanation about the features of the three interfaces and the procedures of the tasks in detail. After this, they were asked to don the headset to start the study.

The participants were exposed to the interfaces in a random order. For each interface, participants had one training trial before the formal tasks started. After all the five formal tasks of each interface were completed, participants were asked to answer questions on a questionnaire on a Likert-scale from 0 (strongly disagree) to 7 (strongly agree). Some of the rating questions were based on the study of Harms and Biocca [HB04], and were used to measure social presence. After the participants finished all the three interfaces trials, they were also required to provide opinions about the advantages and disadvantages of each interface, and choose one of interfaces as they liked to use most. Our questionnaire has been shown in Table 1.

Table 1: *interview questions*

	Likert-scale Questionnaire
Q1	This system was easy to use
Q2	The interface was helpful to complete the task
Q3	I feel confident using this system
Q4	My partner's presence was obvious to me
Q5	My presence was obvious to my partner
Q6	It was easy to understand my partner
Q7	My partner found it easy to understand me
	Post-experiment Questionnaire
Q1	Which interface that you prefer to use
Q2	Benefits of each interface
Q3	Problems of each interface

5. Result and Discussion

5.1. Task Performance

Overall, 98.7% of the trials were completed correctly (only 2 errors of a total of 5*3*10=150 tasks). We noticed that both errors were made by one participant who accidentally identified the orange Lego model as a red model. We conclude that all the participants focused on their tasks during the study, so the task completion time was meaningful. With a one-way repeated measures ANOVA, we found no significant difference between interfaces on the task completion time with F=1.245, p=0.304. In this case, all the three interfaces have almost the same positive effects on supporting collaborative tasks. Figure 7 shows the average task completion time with standard error.

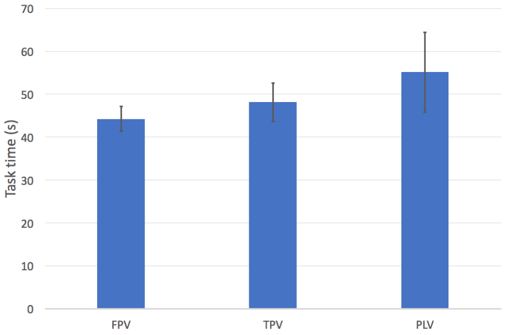

Figure 7: *Average task completion time*

5.2. Questionnaires

Participants were asked to provide their subjective feedback on some rating questions (Table 1) immediately after the trials for each interface. To compare the Likert-scale ratings between the three interfaces, we ran the Friedman test (α=0.05). For those results showing a significant difference between the three conditions, we ran post hoc tests for pairwise comparison using the Wilcoxon Signed-Rank test with Bonferroni correction applied (α=0.0167).

According to the Friedman test, only the question "This system was easy to use" (Q1) shows a significant difference between the three interfaces ($\chi^2(2)$=8.308, p=0.016). However, the Wilcoxon Signed-Rank test indicated that there were no significant pairwise difference (FPV and TPV: Z=0.000, p=1.000; FPV and PCV: Z=-2.266, p=0.023; TPV and PCV: Z=-2.232, p=0.026). For all the other questions, no statistically significant difference was found between the three interfaces (Q2: $\chi^2(2)$=3, p=0.223; Q3: $\chi^2(2)$=4.846, p=0.089; Q4: $\chi^2(2)$=4.522, p=0.104; Q5: $\chi^2(2)$=5.120, p=0.077; Q6: $\chi^2(2)$=1.652, p=0.438; Q7: $\chi^2(2)$=5.083, p=0.079).

In the post-experiment questionnaire, 50% of the participants chose the FPV interface as the one they prefered to use most. Statements by participants about their choices included: "it was easy to check what the partner was working on", "the other person's presence was obvious to me" and "this interface requires less physical movements in order to find the correct objects". Thirty percent of the participants selected the TPV interface as their first choice because: "the view screen is bigger than the first person view and image is more clear" and "this interface makes me feel presence in the scenario". Only two users out of ten said they preferred to use the PCV interfaces. They thought the PCV interface had some unique advantages, such as: "the 3D point is direct and specific" and "it is very easy to find the target label in the first place".

5.3. Discussion

To summarize the results, all the three interfaces supported the remote expert on guiding the local worker to complete some physical tasks in the large workspace. However, no statistical difference was found in the average task performance for each of the interfaces. Based on the feedback of post-experiment questions, we believe that all the three interfaces have their own advantages and disadvantages.

In terms of the FPV interface, participants provided a positive feedback about seeing the 3D background and 2D foreground at the same time. They thought it was straightforward for them to check the real-time feedback from the local worker through a 2D video window that always followed their view. They had the ability to monitor the changes of the local workspace through the duration of the tasks with less physical movement. However, some of them also mentioned that the first-person view window was small, and it was sometimes hard to see the view clearly.

For the TPV interface, as the 2D video window was following the local worker's viewpoint movement, users thought this interface was helpful for them to understand their partners' actions in the scene. It was easy for them to confirm whether or not their partner was following the right order. Furthermore, the 2D video window of this interface was larger and more clear than the FPV interface. However, the remote expert had to move to check the video view since they could only see it behind the local worker's virtual view frustum. Another issue was that this third-person view sometimes blocked the expert's view of the 3D background in the VR space, and interrupted the expert guidance.

Users felt the representation of the PCV interface was more natural than the other two interfaces. In the remote expert's VR world, local changes were displayed exactly the same as where they took place in the local workspace. In this case, it could support better spatial awareness for the remote experts. One limitation reported by the users was that they were not sure whether or not the local worker was looking at the right model until the worker moved the model. While using the PCV interface, the users could only know where the local worker was looking at when changes happened in the local scene. Otherwise, users needed to guess the worker's view area based on the worker's view frustum. Furthermore, due to the narrow field of view of the depth sensor, it took more effort for the experts to guide the local workers in finding the right target.

6. Conclusion and Future Work

In this paper, we have presented a prototype remote guiding system with a hybrid interface combining together a low-resolution 3D point-cloud scene with a high-resolution real-time view. In this case, the remote expert had an overview of the local workspace with independent viewpoint control. Meanwhile, they could also check the local worker's actions based on the real-time 2D video or 3D point cloud feedback.

We designed three interfaces to evaluate the performance of our system while working in a large work environment. We did not find any task performance differences between these three conditions. The different presentation of real-time 2D videos or 3D point cloud had their own advantages and disadvantages to assist remote expert on guiding the local worker.

In the future, based on the user feedback, we plan to alter our current remote collaboration system to not show the static local scene and real-time view feedback at the same time. As an alternative approach, we may let the user to choose which view he/she needs at a given time. For example, when the user wants to check the general geometric layout of the local workspace, he/she could switch to the static point cloud background view. On the other hand, if the user wants to see the local worker's actions, he/she could change to the real-time foreground view. Furthermore, we also want to provide real-time view feedback not only from the local worker's point of view, but also from other viewpoints, such as a side view or a God's eye view. World stabilized remote annotation is also one research area we want to explore in our future studies.

References

[AAT13] Adcock M., Anderson S., Thomas B.: Remotefusion: real time depth camera fusion for remote collaboration on physical tasks. In *Proceedings of the 12th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry* (2013), ACM, pp. 235–242. 2

[ABM15] Amores J., Benavides X., Maes P.: Showme: A remote collaboration system that supports immersive gestural communication. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems* (2015), ACM, pp. 1343–1348. 3

[AL11] Alem L., Li J.: A study of gestures in a video-mediated collaborative assembly task. *Advances in Human-Computer Interaction 2011* (2011), 1. 1, 2

[BNR14] Billinghurst M., Nassani A., Reichherzer C.: Social panoramas: using wearable computers to share experiences. In *SIGGRAPH Asia 2014 Mobile Graphics and Interactive Applications* (2014), ACM, p. 25. 2

[CM91] Chen Y., Medioni G.: Object modeling by registration of multiple range images. In *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on* (1991), IEEE, pp. 2724–2729. 5

[FKS00] Fussell S. R., Kraut R. E., Siegel J.: Coordination of communication: Effects of shared visual context on collaborative work. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work* (2000), ACM, pp. 21–30. 2

[FSK03] Fussell S. R., Setlock L. D., Kraut R. E.: Effects of head-mounted and scene-oriented video systems on remote collaboration on physical tasks. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2003), ACM, pp. 513–520. 1, 2

[GLB16] Gupta K., Lee G. A., Billinghurst M.: Do you see what i see? the effect of gaze tracking on task space remote collaboration. *IEEE transactions on visualization and computer graphics 22*, 11 (2016), 2413–2422. 1

[GNTH14a] Gauglitz S., Nuernberger B., Turk M., Höllerer T.: In touch with the remote world: Remote collaboration with augmented reality drawings and virtual navigation. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology* (2014), ACM, pp. 197–205. 2

[GNTH14b] Gauglitz S., Nuernberger B., Turk M., Höllerer T.: World-stabilized annotations and virtual scene navigation for remote collaboration. In *Proceedings of the 27th annual ACM symposium on User interface software and technology* (2014), ACM, pp. 449–459. 2

[GXS*12] Giusti L., Xerxes K., Schladow A., Wallen N., Zane F., Casalegno F.: Workspace configurations: setting the stage for remote collaboration on physical tasks. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design* (2012), ACM, pp. 351–360. 2

[HB04] Harms C., Biocca F.: Internal consistency and reliability of the networked minds measure of social presence. 7

[JGM15] Johnson S., Gibson M., Mutlu B.: Handheld or handsfree?: Remote collaboration via lightweight head-mounted displays and handheld devices. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing* (2015), ACM, pp. 1825–1836. 2

[KLSB14] Kim S., Lee G., Sakata N., Billinghurst M.: Improving co-presence with augmented visual communication cues for sharing experience through video conference. In *Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on* (2014), IEEE, pp. 83–92. 1, 2

[KNR14] Kasahara S., Nagai S., Rekimoto J.: Livesphere: immersive experience sharing with 360 degrees head-mounted cameras. In *Proceedings of the adjunct publication of the 27th annual ACM symposium on User interface software and technology* (2014), ACM, pp. 61–62. 2

[KRF07] Kirk D., Rodden T., Fraser D. S.: Turn it this way: grounding collaborative action with remote gestures. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (2007), ACM, pp. 1039–1048. 1

[LSCG13] Lanir J., Stone R., Cohen B., Gurevich P.: Ownership and control of point of view in remote assistance. In *Proceedings of the*

SIGCHI Conference on Human Factors in Computing Systems (2013), ACM, pp. 2243–2252. 2

[RBB07] RANJAN A., BIRNHOLTZ J. P., BALAKRISHNAN R.: Dynamic shared visual spaces: experimenting with automatic camera control in a remote repair task. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2007), ACM, pp. 1177–1186. 2

[SFG*16] SILVA R., FEIJÓ B., GOMES P. B., FRENSH T., MONTEIRO D.: Real time 360° video stitching and streaming. In *ACM SIGGRAPH 2016 Posters* (2016), ACM, p. 70. 2

[SJF*13] SODHI R. S., JONES B. R., FORSYTH D., BAILEY B. P., MACIOCCI G.: Bethere: 3d mobile collaboration with spatial input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2013), ACM, pp. 179–188. 2

[SKY*12] SEO D., KIM S., YOO J., PARK H., KO H.: Immersive panorama tv service system. In *Consumer Electronics (ICCE), 2012 IEEE International Conference on* (2012), IEEE, pp. 201–202. 2

[TAH12] TECCHIA F., ALEM L., HUANG W.: 3d helping hands: a gesture based mr system for remote collaboration. In *Proceedings of the 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry* (2012), ACM, pp. 323–328. 3

[TL08] THRUN S., LEONARD J. J.: Simultaneous localization and mapping. In *Springer handbook of robotics*. Springer, 2008, pp. 871–889. 2