A Conceptual Framework of Delay Perception in 3D Tele-Immersion

Leave Authors Anonymous
Institute
City, Country
example@email.com

Leave Authors Anonymous
Institute
City, Country
example@email.com

Leave Authors Anonymous
Institute
City, Country
example@email.com

ABSTRACT

3D Tele-Immersion (3DTI, e.g., Holoportation [73]), allows distributed users to communicate and interact with each other in the same virtual space. It grows rapidly in recent years. However, no work has studied network delay perception in 3DTI. Network delay is an important factor that affects user experience. In this paper, we explore users' perception of network delay in 3DTI. We propose a conceptual framework that classifies 3DTI tasks into three levels by their network delay requirement: *synchronous tasks*, *turn-based audiovisual tasks* and *turn-based visual-only tasks*. They require a network delay within about 50 ms, 250 ms, and 300 ms respectively. The framework introduces how the users perceive network delay in 3D. There are two tendencies of 3D delay perception. First, much more applications fall into the first level *synchronous tasks* in 3DTI. We should pay attention to support their very low network delay. Second, the users are less sensitive and more tolerable to the network delay of *turn-based audiovisual tasks* in 3D. Based on these findings, the framework gives suggestions for network design. Finally, we describe a controlled study as illustrating examples and validation.

CCS CONCEPTS

• **Human-centered computing** → *HCI theory, concepts and models*;

KEYWORDS

Delay perception; 3D tele-immersion

ACM Reference Format:
Leave Authors Anonymous, Leave Authors Anonymous, and Leave Authors Anonymous. 2019. A Conceptual Framework of Delay Perception in 3D Tele-Immersion. In *Proceedings of ACM SigCHI conference (CHI'19).* ACM, New York, NY, USA, 18 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

The past centuries have witnessed the growth of communication technology. The invention of the telephone has saved a great deal of time and money by displacing physically face-to-face meeting. In recent two decades, 2D audiovisual communications are getting popular, such as teleconference [1, 64], telecollaboration [2, 20], robotic telepresence [43, 66, 67], and so on.

Immersion is a tendency in the development of communication technology. 3DTI emerged in the past decade [54, 61, 62, 78]. They are developing toward the highest level of immersion that allows distributed users to communicate and interact with each other in the same virtual space. Both the improvement of pipeline and hardware make 3DTI hopeful to be practical in the near future. Microsoft's Holoportation [73] is a typical 3DTI pipeline with high-quality, real-time performance. For computing, GPUs are getting more powerful. For rendering, immersive displays such as Head-Mounted Displays (MHDs) are becoming popular.

Network delay is one of the critical problems in communication technology. On the one hand, new communications challenge the improvement of network delay, because they generally require higher bandwidth [49]. On the other hand, network delay is a crucial factor that affects user experience [8, 84, 86, 96]. For example, the delay of 150 ms provides a good user experience for most audio-mediated applications [20, 81]. It has become an industry standard that contributes to telephone network engineering [38].

Numerous works have been carried out to explore delay perception in telephone and 2D communications. However, no work has been done to study users' perception of network delay in an advanced 3DTI system, i.e., with reconstruction and rendering in full 3D. It is necessary to rebuild the framework of delay perception in 3D because there is a big difference between 2D and 3D. First, an advanced 3DTI system support co-present, that is, the two users feel like exactly at the same virtual space. Second, 3DTI offers much more visual information. These new features may lead to unknown changes of delay perception in 3D.

In this paper, we explore users' perception of network delay in a full 3D tele-immersion system. We systematically reviewed previous works on delay perception in 2D. Previous works inspire us so that we had several hypotheses about the problem. To validate the hypotheses, we first built a 3DTI system with an ideal network delay of 50 ms, and

CHI'19, April 2019, ElPaso, Texas USA
2019. ACM ISBN 123-4567-24-567/08/06…$15.00
https://doi.org/10.475/123_4

then conducted a controlled study. Finally, we proposed a conceptual framework of network delay perception in 3DTI.

The framework suggests that users perceive network delay by cues. Users can perceive the network delay if they feel that his partner response to the cues abnormally, because the perceived response time is mixed with the network delay. According to previous works, there are mainly three types of cues: synchronous actions, conversation and visual feedback. Among these cues, synchronous actions are the strongest cue to reveal a network delay, while the visual feedback is the weakest one. Thus, we classifies 3DTI tasks into three levels: *synchronous tasks*, *turn-based audiovisual tasks* and *turn-based visual-only tasks*, which require network delays within about 50 ms, 250 ms and 300 ms.

The framework infers significant changes of network delay perception in 3D. First, much more applications fall into the first level *synchronous tasks* in 3DTI. We should pay attention to support their very low network delay. Second, the users are less sensitive and more tolerable to the network delay of turn-based audiovisual tasks in 3D. Based on these findings, we give suggestions on network design of 3DTI to improve the user experience.

Our contribution is fourfold: first, the framework infers significant changes of network delay perception in 3D. We recommend 3DTI developers to assess their applications through our framework, in order to design the transmission properly. Second, we describe two tasks in the study for illustrating examples and the validation of our framework. Third, we give suggestions on network engineering for each level of task. These suggestions can help saving network resource and improving the user experience. Fourth, our project is open-source [?]. We give the necessary explanation in the system overview to make sure that the readers can easily build up a similar system.

In the remainder of the paper, we first present our framework (section 2). We next give an overview of our experimental 3DTI system (section 3). We then describe the controlled study to illustrate our framework (section 4). We supplement related works on 3DTI systems and existing studies on 3DTI delay perception (section 5). The paper concludes by discussing our limitation and the future work (section 6).

2 A CONCEPTUAL FRAMEWORK OF DELAY PERCEPTION IN 3DTI

The framework classify possible 3DTI tasks by their requirement of network delay. There are three levels: *Synchronous* tasks, *Audiovisual* tasks and *Visual-only* tasks, which require networks delay of about 50 ms, 200 ms and 300 ms respectively.

We recommend the 3DTI practitioners to first assess the level of their application, because the delay requirements vary a lot in different levels of tasks. Furthermore, the framework gives practical suggestions of network engineering for each level.

The three levels are described as follow:

- **L1: *Synchronous* tasks (50 ms).**
  The actions in two sides are at the very same time. The users can perceive the network delay by judging if the actions are synchronous in both audio and visual channels. For example, *shaking hands*, *the Rock-Paper-Scissors game*, *instrument ensemble* and *dancing together*.
- **L2: *Audiovisual* tasks (200 ms).**
  The interactions are in turn. The audio channel and the visual channel are both available. The users can perceive the network delay by judging if the conversation and the visual feedback are in time. For example, *free conversation*, *tele-collaboration* and *video conference*.
- **L3: *Visual-only* tasks (300 ms).**
  The interactions are in turn. Only the visual channel is available. The users can only perceive the network delay by the visual feedback. For example, *playing chess silently* and *surgery simulation*.

In this section, we first propose the foremost deduction in short. Next, we review previous works on 2D delay perception for preparation. Then, we have three subsections to introduce the three levels in details. These subsections explain our deduction. Last, we gives suggestions on network engineering for each level.

Background in 2D

In 2D, most applications are *Audio-only* and *Audiovisual* interactions in turn. [?, ?, ?] suggested that delay of 150 ms to 200 ms can support most tasks in audio and 2D DIME systems. For comparison, *Synchronous* tasks are rare because of the lack of spacial togetherness in 2D. There are also few *Visual-only* tasks because 2D visual information is not sufficient for communication [?].

In summary, the requirement of network delay among tasks in 2D is shown as figure 1.

The figure is summarized by a comprehensive review on delay perception in 2D. In general, an user study only focus on a specific task. The method is to collect users' subjective rating of different network delay via questionnaires. Mean Opinion Score (MOS) is the average subjective rating on a 5-point Likert scale [39, 80]. It is measured by most studies [84, 85, 93, 96]. 3.5 MOS is regarded as a threshold that can provide a good user experience [22, 83]. Thus, we summarize the 3.5 MOS threshold in previous works to fill the figure above. However, some previous works did not measure MOS. In this situation, we regarded the network delay that can

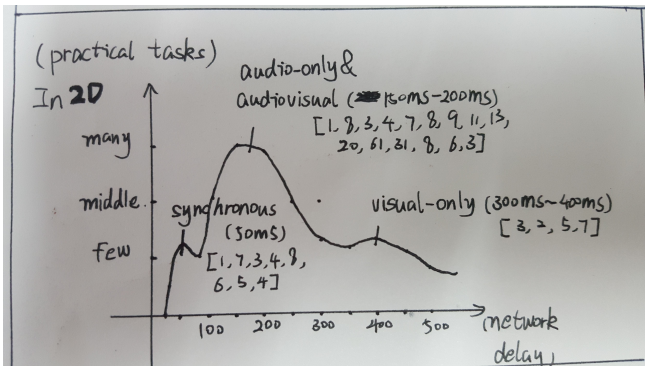

Figure 1: The distribution of network delay requirement among tasks in 2D. The ordinate value is an empirical assessment based on paper survey.

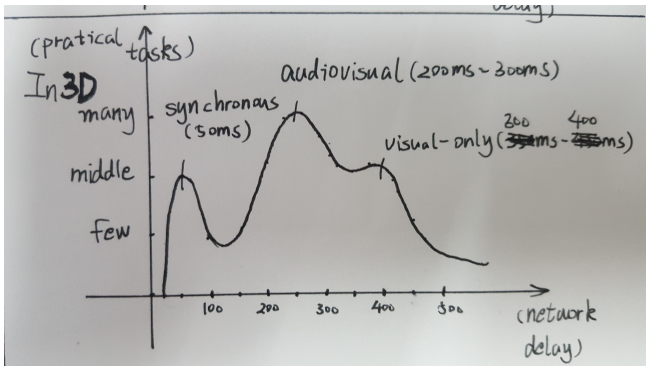

Figure 2: The distribution of network delay requirement among tasks in 3D. The ordinate value is an empirical assessment based on our framework.

provide a good user experience as the recommended threshold.

The Foremost Deductions in 3D

The distribution of network delay requirement among tasks change a lot in 3D. The three most significant deductions are:

- **D1: Much more *synchronous* tasks.**
  Previous works have found that a few tasks involve synchronous interaction, which require a very low network delay of about 50 ms. Unfortunately, we suggest that 3DTI supports much more *synchronous* tasks or improves them in a natural manner. These tasks are extremely challenging to network engineering.
- **D2: The decreased sensitiveness and increased tolerance to *audiovisual* tasks.**
  Previous works suggest that delay of 100 ms to 150 ms is enough to support most applications in 2D [?]. Fortunately, we suggest that users are less sensitive and more tolerable to these tasks in 3D. A delay of about 200 ms is acceptable for most tasks in 3DTI. Thus, the 3DTI practitioners have more room to improve the user experience.
- **D3: More *Visual-only* tasks.**
  There are few *Visual-only* tasks in 2D. Most applications in 2D involve the assistance of audio channel, because 2D video is not sufficient for communication. 3DTI provides more visual information so that it supports some silent tasks. Without conversation, the users are further less sensitive to the network delay.

In summary, the requirement of network delay among tasks in 3D is shown as figure 2. The distribution becomes differentiated. Thus, it is valuable to first assess the level of an application. Then we can improve the network engineering accordingly.

1. Synchronous tasks

Synchronous tasks are the interactions that a pair have to act at the very same time. The user can perceive the network delay by judging if the actions are synchronous.

First, we analyze the network delay requirement of *Synchronous* tasks. In a *Synchronous* task, because of the synchronous actions, users' ability to perceive the network delay is nearly the same as the perception of a local system delay. As [70] explained, 100 ms is an upper boundary for users to fell that the system is running instantaneously. For a better performance, a local delay of 30 ms to 50 ms is needed [13]. Thus, we deems that the network requirement of *Synchronous* tasks is about 30 ms to 100 ms. Take the networked *Rock-Paper-Scissors* game for example. If the network delay is 100 ms, one of the players will find that his partner gesture at least 100 ms slower. A delay of 100 ms is very obvious in this game, which may cause annoyance.

Many *Synchronous* tasks require one of the users to synchronize the actions by gesturing or saying something, e.g., "Three, two, one, go!". As [34] revealed, there is a difference between the delay perceptions of the *caller* and the other user *replier*. The *caller* is the user who try to synchronize the actions. As figure 3 shown, the *replier* just adjust the timing of his motion to the *caller's* timing. Thus, he do not perceive any network delay. In contrast, the *caller* experiences the round-trip delay before seeing the partner's reaction. The round-trip delay is about 200 ms in a networked *Rock-Paper-Scissors* game with delay of 100 ms, which is totally intolerable for most users. Thus, we should balance the perceived network delay of the the two users in a *Synchronous* tasks. We will discuss the method in the suggestion part of the framework.

Next, we discuss about **D1**: why 3DTI can support much more *Synchronous* tasks or improve them in a natural manner? The main reason is that 3DTI allows co-present, that is,

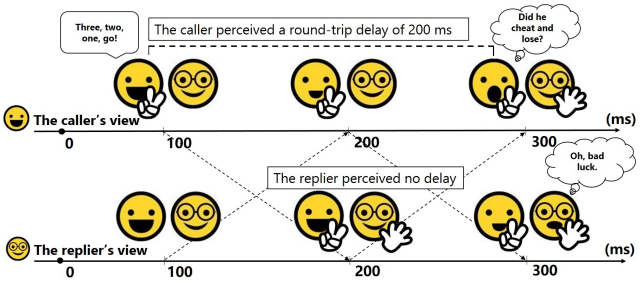

Figure 3: In a networked *Rock-Paper-Scissors* game with a delay of 100 ms, the *caller* is the player who try to synchronize the game, e.g., by saying "Three, two, one, go". He perceives a round-trip delay of 200 ms, while his partner perceives no delay.

the two users feel like exactly at the same virtual space. Co-present is a critical factor in a full 3D tele-immersion system [51, 73, 94]. In previous audiovisual DIME and the 3DTI systems without HMDs, the interaction always occurs through a 'window' from one space into the other. Co-present allows the users to "touch" each other, exchange their position freely and use the shared physical props [60]. Correspondingly, a 3DTI system supports hand shaking, dancing together and piano duet. These tasks relied on co-present tend to involve *Synchronous* interactions, which require a low network delay.

Then, we introduce some examples in the *Synchronous* level:

Shaking hands.

Counting down together.

Musical collaboration. Development of audio transport over networked have supported professional-quality musical collaboration [10, 11]. The emerging 3DTI may support a vivid musical collaboration in the future. As [87] suggests, the threshold of the network delay requirement of musical collaboration lies between 20 ~ 30 ms. With a specific coping strategy (a leader / follower relationship), the musicians can maintain the performance up to 50 ~ 70 ms.

Dancing together.

2. Audiovisual tasks

Audiovisual tasks are the turn-based interactions with the audio channel and the visual channel available. Notice that this level excludes *Synchronous* tasks. The user can perceive the network delay by judging if the conversation or the visual feedback is slower than in the real communication.

First, we review the theoretical background of the delay perception in *audiovisual* tasks. In a *Audiovisual* task, the users perceive network delay by conversation and visual feedback. For a communication, the audio channel is sufficient [94]. The visual channel is an assistance, though it offers much more information than audio. In general, conversation is a stronger cue for delay perception compared to visual feedback.

There are two theories refer to the delay perception of *audiovisual* tasks: Turn-Talking Model and Grounding Theory. Turn-Talking Model is a theory on understanding conversation, while Grounding Theory explains how visual information improve verbal communication.

Turn-Talking Model. Turn talking is a part of universal infrastructure for language [55]. In daily life we have learned to unconsciously manage a conversation by using the timing of the small pauses in speech [82]. Figure 4 shows the typical timing of the conversation. A Turn is 2 s in average. The language production system is slow: preparation before output begins takes 600 ms to 1500 ms [3, 31, 36]. However, switching of speakers is rapid, because the turn talking system relies on prediction [55]. The modal response time (gaps between turns) is around 200 ms [56].

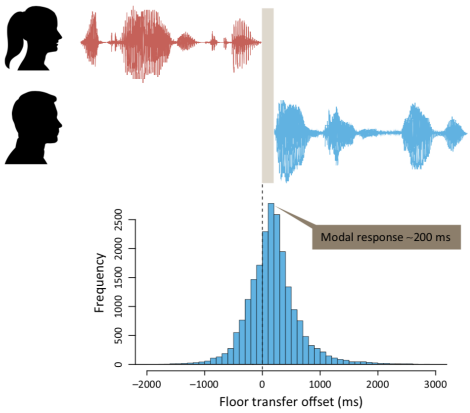

Figure 4: [注意，这是盗图] Turn Talking Model.

The short response time in conversation make the users sensitive to the network delay, because the network delay prolong the perceived response time. Delay of 100 ms can be noticed in audio telecommunications [38, 81, 93]. Delay of 150 ms becomes industrial standard of the telephone network [81]. Delays greater than 450 ms can severely impact communication [28, 72, 90]. In short, conversation is an important cue for the users to perceive network delay.

Grounding Theory. Grounding Theory suggests that a successful communication relies on a foundation of mutual knowledge or common ground [14, 15]. Shared visual information is an important source of common ground that provides evidence of comprehension for tele-communication

[7, 51]. Thus, sufficient visual information can assist the verbal communication [28] and reduce users' dependence on conversation. For example, a user can point at an object in the shared virtual space and refer to it using a simple pronoun "that". [NOTE] so what? Grounding Theory for delay perception.

Next, we discuss about **D2**: why the users are less sensitive and more tolerable to the network delay in 3D *Audiovisual* tasks? In 2D, video weaken the negative impact of delay in remote interactions, because audiovisual interaction allows users to see visual information [89]. We deems that this effect is enhanced in 3D.

According to Turn-Talking Model and Grounding Theory, the users are more sensitive to the delay of conversation but not visual feedback. Visual information reduce users' dependence on audio communication, so the sensitiveness of delay is decreased accordingly. As [51, 52] suggests, when all parties to the interaction are co-present, the users share a rich visual space. Thus, 3DTI systems with co-presence provide the richest visual information. It enhances the effect that the users are less sensitive and more tolerable to the network delay.

Then, we introduce some examples in the *Audiovisual* level:

Video Conference.

Free Conversation.

Tele-collaboration. For example, fixing a bike.

3. Visual-only tasks

Visual-only tasks are the turn-based interactions with only the visual channel available. This level also excludes *Synchronous* tasks. The user can perceive the network delay by judging if his partner acts slower than in the real communication.

First, we review the Situation Awareness Theory to help understanding how the users perceived network delay by visual feedback. Situation Awareness Theory holds that visual information helps pairs assess the current state of the task and plan future actions [23, 24]. A user need to maintain an on-going awareness of the partner' s actions and the status of the task objects. Through the observation of the respond time, a user may perceive the network delay. However, the response time of an visual feedback is harder to predict compared to the gap in a conversation. Thus, users are less sensitive to the network delay in a *Visual-only* task.

Previous works shows that the network delay requirement of the *Visual-only* tasks is generally looser, from xxx ms to xxx ms [?, ?, ?].

Next, we discuss about **D3**: why 3DTI can support more *Visual-only* tasks?

[NOTE] I think that **D3** is a truth, but not useful. So it may be deleted latter.

Then, we introduce some examples in the *Visual-only* level:

Surgery Simulation.

Playing Chess.

Suggestions for network design

There is a space to improve the network design for a 3DTI system, because both the system service and the user experience in 3D are different from the 2D situation. On the one hand, the computation is tough in 3D, which leads to a generally lower frame rate. The bandwidth requirement is also larger (1.5-fold of a 1080p video [?]); On the other hand, users' perception of network delay change a lot in 3D. The practitioners can suit their methods to the situation when designing the networked applications:

- Zero-delay audio assistance (for **L1**):
  In *Synchronous* tasks, the users are sensitive to the network delay. Moreover, as we explained above, the *caller* may perceive a round-trip network delay. To help synchronizing the task, we suggest to add a zero-delay audio prompt tone in the application. In the networked *Rock-Paper-Scissors* game, for example, we can add synchronized sounds of "tick, tick, tack" for both the users. The users can show their gesture when they hear the "tack" sound so that the actions can be exactly at the same time. The advantage of this strategy is twofold: first, to avoid the perception of round-trip network delay; second, the psychological hint that the game is fair.
  Though the network delay is unavoidable in the audiovisual transmission, it is possible to accurately synchronize the timing of two systems through the Network Time Protocol (NTP) [65]. Thus, this strategy is practicable. In our experiment, we validated that this strategy can significantly improve the user experience.
- Trade bandwidth for time (for **L1**):
  In general, delay and bandwidth are a trade-off in network transmission. Compression is the key. It reduce the bandwidth requirement of a 3DTI network. Recent works on 3D data compression shows that a bandwidth of tens of megabits per second is possible to support a 3DTI system [16, 19]. However, compression is time-consuming. Some compression methods are based on inter prediction, which leads to several frames of latency.
  The *Synchronous* tasks require a network with delay of 50 ms. The service has almost no time for compression. In this situation, we can trade bandwidth for time,

i.e., to use a lightweight compression method, or even transmit the raw data directly (about 500 Mbps). This strategy can reduce the network delay as well as maintain the highest quality. The price of this strategy is the very high requirement of network bandwidth. Thus, it is only possible if the practitioners have a dedicated network.

- Buffer frames to recover lost packets (for **L2** and **L3**)
  Delay is not the only factor that affects the user experience in a networked communication. Jitters, delay spikes and network losses will degrade the user experience as well. Fortunately, the network requirement of turn-based *Audiovisual* tasks and *Visual-only* tasks are looser in 3D. We suggest that the acceptable threshold in 3D is 50 ms ∼ 100 ms larger than that in 2D. Thus, we can increase the network delay to within the threshold, in order to have more room for smoothing data stream and recovering lost packets.
- Trade time for bandwidth (for **L2** and **L3**)
  As we explained above, delay and bandwidth are a trade-off. For turn-based *Audiovisual* tasks and *Visual-only* tasks, we can trade time for bandwidth, i.e., to use a heavyweight compression method. In the state-of-the-art 3D reconstruction pipeline [21, 73], it is important to track the 3D model based on temporal consistency. It also provide convenience for the compression of data stream, i.e., we can introduce predictive-frame in the transmission.
- Transmit the audio faster than the video (for **L2**)
  In a turn-based *Audiovisual* task, the network service to support the audio channel is more lightweight than that to support the video channel. Coincidentally, small delay can seriously disrupt the audio communication [50], while the delay requirement of the video is looser. However, most video conference systems synchronize video and audio by delaying audio, which reduces the responsiveness of the conversation [?, ?, ?, ?]. [37] suggests that we can transmit the audio a little bit faster than the video. Even if the gap between audio and video is noticeable, this strategy can somehow improve the user experience. In the 3D situation, we suggest a gap between the audio channel and the visual channel within 100 ms is acceptable.

LZP's Review

Four levels

1.Synchronous Interaction

Typical cases of synchronous interaction are rock-paper-scissors game and virtual car driving. In a virtual car driving task [74], the driver must react as soon as a situation appears. In this way, a task like virtual car driving is a kind of synchronous interaction. As for rock-paper-scissors game, fairness is the most important part and the only way to guarantee fairness is to make sure both players show their hands meantime. However, the experience of two players in rock-paper-scissors can be different. Generally, the more active participant feels unfair while the other one feel nothing. This strange situation is explained in [34].

2.Conversation only

There are many cases in daily life using audio remote interaction. And there are many researches about it. So I find several tasks in [48] to provide data.

3.audiovisual

这段用英文我觉得可能说不清楚。我个人理解是这样的，对于audiovisual的task，visual只是audio的辅助而已。因为在2D中，visual能提供的信息还是太少了，虽然看起来和打电话这样的纯audio相比，的确提供了视觉信息，但是这些2D的视觉信息并不能真正起到什么作用，只是提供给使用者一个图像而已，主要的交流还是用语音的。因此visual只是audio的辅助。所以我觉得大部分的audiovisual的task都是偏audio的，少数偏visual的task是类似你画我猜这样的task，限制一方的语音使用，只能通过visual来传递信息。

从论文中的结果来看，audiovisual的task略微比audio only的延迟容忍要高，这个部分的原因我觉得是因为visual信息分散了使用者的注意力，也就对延迟上不是那么敏感。这只是一个我个人主观上的猜测。

在这里和上面对比，我大概说一下我认为3D中visual信息能起到的作用。在3D中，视觉信息比2D中要多得多，比如在2D中我们只能看到对方的手的一个角度，而在3D中我们可以看到手的任意部位任意角度，因此在2D中可以在特定角度下玩石头剪刀布，而在3D中就可以握手甚至更多身体"接触"，因此3D中的visual信息是很重要的，就算只有visual没有audio，在3D中可以传达的信息也是很多的，比如通过用自己的行为进行亲身示范，或两个人共同操作同一个物品，或者通过3D来进行一些训练例如驾驶，我们完全可以想象就和在现实中一样，只是不允许说话。那我们现实中的所有场景或task理论上都可以在3D中进行。

audiovisual论文总结

—

title:Understanding Performance in Coliseum, An Immersive Videoconferencing System

内容:和我们类似的三维重建，2005年实现的，质量很差，只是测量了延迟和不同部分的延迟比例，做到了200-250ms的延迟

—

title:Exposure to asynchronous audiovisual speech extends the temporal window for audiovisual integration

内容:听觉和视觉的不同步并不容易察觉到，做了实验但没有给出具体延迟阈值

还有多篇论文提出一样的结论，但是数据不尽相同，大概在100ms左右

—

title:The Effect of Interactivity on Learning Physical Actions in Virtual Reality

内容:VR中的沉浸式学习，task都是肢体动作的学习和模仿，给出了instructor的镜像和本人两种画面

—

title:Effects of Head-Mounted and Scene-Oriented Video Systems on Remote Collaboration on Physical Tasks

内容:通过对比试验，提供不同层次的信息，task是两人协作操作机器人，分别是side-by-side,audio-only, head-mounted camera, scene camera, and scene plus head cameras。结论是scene camera与现实中的side by side最接近，打分最高

—

title:Real-time Terascale Implementation of Tele-immersion

内容:描述在Pittsburgh Supercomputing Center实现的场景重建，提供沉浸式体验

4.only visual feedback

纯粹的visual在2D中真的不多，大部分都是带有audio的，因为如果没有语音通话，实验者只通过visual很难有效传达信息或者表述，所以这样的task几乎没有。

Table 1 shows the review of previous work.

HBJ's Theory

We present an empirical framework of delay perception of 3D Tele Immersion. Tasks in 3DTI can be assigned to 3 different levels: tasks with enforced synchronous interaction, tasks with audiovisual conversation and tasks with visual-only communication. Tasks are divided into three levels according to their different requirements of delay, as shown in figure 1(表格)

There are two indicators of delay measured by most researches: noticeability and tolerance of delay. [89]

Noticeability Noticeability of delay means whether user can perceive the transmission delay in the system. In common case, if they can communicate very fluently in the visual space as if they were communicating face to face in the real world, then can barely notice the delay.

Tolerance Tolerance means whether user can tolerate the delay in the system or not. If they believe the delay hinder them from interacting fluently with each other or disrupt the fairness of certain task and game, the tolerance will be very conspicuous.

In our relevant experiments (discussed later in this article), we use Mean opinion score (MOS) to measure delay noticeability and tolerance. MOS is a measure used in the domain of QoE by many researchers to evaluate subjective opinions.[9, 39, 93, 96] It is expressed as a single number on a scale of 1 to 5, where 1 is the lowest perceived quality and 5 is the highest. The final result is the arithmetic mean of scores rated by all experimental subjects. According to industrial standard, a MOS score over 3.5 indicates that the system is satisfying. We employed the standard in our

Levels and *Examples*	Noticeable Delay	Tolerable Delay	3.5 MOS
Synchronous Interaction	**20 - 50 ms**	**50 - 100 ms**	?? ms
Rock-Paper-Scissors [34]	40 ms	?? ms	70 ms
Virtual car driving [74]	50 ms	200 ms	?? ms
example [?]	?? ms	?? ms	?? ms
Conversation only	**100 - 150 ms**	**300 - 400 ms**	?? ms
Take turns reading random numbers aloud as quickly as possible [48]	?? ms	?? ms	80 ms
Take turns verifying random numbers as quickly as possible [48]	?? ms	?? ms	120 ms
Take turn verifying city names as quickly as possible [48]	?? ms	?? ms	180 ms
Free conversation [48]	?? ms	?? ms	200 ms
example [?]	?? ms	?? ms	?? ms
audiovisual	**100 - 150 ms**	**300 - 400 ms**	?? ms
3D Visual Communication [96]	120 ms	?? ms	?? ms
Video Group Discussion [84]	500 ms	1000 ms	500 ms
Audiovisual telecommunication [89]	?? ms	?? ms	500 ms
example [?]	?? ms	?? ms	?? ms
Visual Feedback	**150 - 500 ms**	**500 - 1000 ms**	?? ms
example [?]	?? ms	?? ms	?? ms
example [?]	?? ms	?? ms	?? ms
example [?]	?? ms	?? ms	?? ms
example [?]	?? ms	?? ms	?? ms
example [?]	?? ms	?? ms	?? ms
example [?]	?? ms	?? ms	?? ms

Table 1: The three synchronization levels.

experiments as well. Each subject is required to give their rating of delay. If the MOS score of noticeability is over 3.5, that means subjects can't perceive delay. If the MOS score of tolerance is over 3.5, that means subjects can tolerate delay.

Countless work has been done on measuring these two factors in previous researches.[11, 34, 89, 93] However, all of them were conducted in audio-only experiments and 2D video experiments, as shown in figure 2(表格介绍以往工作). In audio-only telecommunication research, it is widely acknowledged that the noticeability and tolerance of delay are around 150ms and 400ms (引用). As for audiovisual experiments noticeability and tolerance of delay are around 200ms and 500ms. (need citation)

However, the framework of delay in 3DTI tasks is different from both of them. According to our deduction, tasks can be divided into three levels. In each level, tasks have similar threshold of delay noticeability due to their common audio and visual factors, while threshold of delay tolerance usually vary in a small range because of other factors related to specific tasks.

Synchronous tasks The common feature of these tasks is that they all require users to correspond their behavior with each other. Users in these tasks tend to notice delay easily through observing partner's behavior. Besides, high delay is usually intolerable because it has obvious negative impact on QoE and fairness with game. Delay noticeability is around 50ms and Delay tolerance varies from 50ms to 150ms.

Typical examples of enforced synchronous interaction are Rock-paper-scissors game, real-time musical collaboration and count-down game. Rock-paper-scissors is a hand game played between two players. In the game, players are required stretch out their hand simultaneously. If delay between two sides grows higher than the threshold of noticeability, user will perceive that his or her partner seems to show gestures later than expected. If it exceeds the tolerance delay, then both players will suspect that their partner shows gestures slowly intentionally and the fairness with the game is broken.[34] Real-time musical collaboration enables distant musicians to perform on the same work. It has an even stricter limitation on delay tolerance because if one of the musician lags behind or go ahead just a little fraction of a musical note (say a quaver), the general harmony of music will be destroyed completely.[11] Thus, noticeability and tolerance are almost the same in this task: 50ms. Count-down game demands that both players to count from 10 to 1 simultaneously. This is a rather low-requirement task of level 1.[93] Players can tolerate delay up to 150ms because they can't count down precisely with partners even in real world.

In addition, we discovered that a synchronized signal for all players can help extends the limit of delay. In our experiment of Rock-paper-scissors game, we gave both players a synchronized audio signal so that they can followed the signal to stretch out their hand simultaneously. Compared with games without signal, results show that players can bear higher latency with the synchronized signal. The delay noticeability and tolerance of Rock-paper-scissors game can be extended to 100ms and 300ms.

Audiovisual tasks The common feature of these tasks is that users can communicate in virtual visual space with audio information. Unlike synchronous tasks, users in these tasks don't need to behave simultaneously, which means it is hard for them to perceive delay by observing partners. Users need special cues to perceive delay in different tasks. The requirement of delay in this level is looser than in the former level. Delay noticeability is around 450ms and Delay tolerance varies from 600ms to 1000ms. Turn talking model and grounding theory.

Typical examples of level 2 are turn talking models, 3D web conference and turn-based strategy games. Real-time strategy game?? Turn talking models refer to a group of tasks in which users talk in turns.[55] For example, users can take turns to read numbers, verify city names and describe fruits with long sentences. Turn talking is a universal infrastructure for all languages. (cite turn talking) It states that the average time for speaker to produce a single word is 600ms but the switching gap of speakers is only 200ms, which means that people start to prepare for the next turn before other people finish the last turn. This phenomenon indicates that people can easily notice delay if it exceeds 600ms because people will perceive that there are longer gaps between turns than in face-to-face talk. The delay noticeability is 450ms and delay tolerance is around 750ms. 3D web conference involves multiple people participating in one discussion. Unlike turn talking situation, the gaps between speakers are even shorter because usually more than one speaker want to take the next turn. Thus, the delay noticeability and tolerance are 400ms and 600ms. Turn-based strategy games are games in which players take actions in turns, such as chess, Reversi, Go and Hearthstone. These games have a common feature that when one player is taking moves or thinking about the next move, other players can do nothing but wait for their own turns. Thus, even if the tolerance is higher than the threshold of noticeability, as long as players don't have direct interaction with each other, they are not able to perceive the impact of delay. The delay noticeability and tolerance of these tasks are 500ms and 1000ms.

Visual-only Interaction The common feature of tasks in this level is that audio communication is not available through the whole process. Users can only use body gestures to interact with each other. These tasks are not possible in former 2DTI systems because users cannot express their meaning fully and fluently through a 2D video screen. However, in 3D visual space, since users can see each other thoroughly in 3D

shapes, observe body gestures more easily and understand each other more quickly. Noticeability and tolerance of tasks in this level is much higher than the previous two levels because users usually work in a slow pace in these "silent" tasks and without audio signals they feel difficult to perceive the transmission delay. As Endsley stated[23], Situation Awareness Theory explains how people use surrounding visual information to project future status and make decision. Delay noticeability is around 750ms and tolerance is more than 1000ms.

Some typical examples are turn-based board games such as chess and Reversi. Chess game involves two players who can play the game without even talking to each other. At the beginning of chess games, players can perceive delay more easily because they move and switch frequently. But as games go on, it takes more and more time for them to think about the next move. The slower the game pace is, the less indicators of delay they can perceive. Few information is useful for chess players to perceive the surrounding, which means there are few cues that reveal delay scale during the game. Delay Noticeability and Tolerance are about 1000ms and 2000ms. Reversi game is similar to chess game but it involves more interaction. Because player of each side only have access to pieces of one color (black or white), so when one side of the player moves, the other side is required to help remove discarded pieces so that new pieces can be placed on. This interaction gives cues to players that help them to notice delay. Delay Noticeability and Tolerance are about 750ms and 1000ms.

We believe the new framework of 3DTI tasks is helpful for network engineering developers. We have several suggestions for them: 1. It is hard to develop systems that meet the delay requirement of level 1. But synchronized signals can help extend the limits and improve users' experience. 2. If your system is already within the threshold of delay noticeability, it is ok to sacrifice some time for improving visual quality such as buffering and recovering package loss as long as the round time doesn't exceed limit. 3. If your system already exceeds delay tolerance, then it's better to sacrifice other things to lower down the delay. Otherwise, users will feel choppy in the visual space.

3 SYSTEM OVERVIEW

Our 3DTI system fuses two distributed scenes into a same virtual space in full 3D. Figure xxx illustrates the functionality of our system. The end-to-end delay is 50 ms, i.e., the time interval between a user acts and his remote partner sees.

In summary, a 3DTI system requires three processes: reconstruction, transmission and rendering [27]. For 3D reconstruction, we applied Truncated Signed Distance Function (TSDF) Volume [18] and Marching Cubes [59]. We did not focus on transmission as [4, 75] did, but used a 10 Gigabit Ethernet connection instead. For rendering, we applied HTC Vive (HMD) because it supports co-present interaction.

The contribution of our system implementation is threefold: xxx. The system consists of inexpensive commercial devices ($ 7000). The project is open-source [?].

Hardware and Software Overview

Hardware. The system consists of two capture sites in the two distributed rooms. At each capture site, we had three depth cameras for capturing, a PC for computing and an HMD for rendering. Realsense D415 (depth cameras) were used to capture a volume of $2m \times 2m \times 2m$. The locating place of each camera and its contribution to 3D mesh are illustrated in Figure xx. Each PC had an Intel i7-7700k CPU and a GTX 1080Ti GPU. HTC Vive was used to present the fused reconstruction of both sides. Ten Gigabit network cards (Intel X520-SR2) were used to connect the two capture sites.

Software. OpenCV was used for camera calibration. CUDA was used for image processing and the kernel algorithm. Unity3D was used to implement the high-level application. It fetches live reconstruction from the kernel and renders it in HTC Vive. Python was used for audio transmission. 视音频同步怎么处理的?

Calibration

Calibration between Cameras. The *camera calibration module* in OpenCV was used to calibrate the cameras. Each pair of cameras took ten snapshots (1080p color images) of a glass-made flat checkerboard. Then, OpenCV aligned their coordinates ($SD < 1pixel$).

Calibration between HMD and Cameras. The HTC Vive was calibrated by setting the original point in its software. We placed the original point of the camera coordinates at the same position by using the checkerboard. Hence, we aligned the HTC Vive with the cameras. This calibration is not necessarily accurate because the users can hardly perceive the error [?]. This step also aligned the coordinates of the two capture sites.

Preprocessing

Depth Processing. The cameras acquired depth images of 640×480 pixels at 30 FPS. The Realsense D415 is based on binocular disparity. Thus, disparity values (instead of depth values) were used in the processing to improve accuracy. We applied median filtering, spatial filtering, hole filling and temporary filtering on the depth images.

Color Processing. The cameras acquired color images of 960×540 pixels at 30 FPS. The exposure settings were manually adjusted. We used one RGB camera as a reference and

warped the other cameras to this reference by white balancing and linear mapping.

Background Removal. The system removed unnecessary background and retained only shared objects and individuals. In the calibration step, we recorded the background as RGBD images. At runtime, we removed pixels that are similar to the background based on thresholds.

3D Reconstruction

3D reconstruction is the kernel algorithm of a 3DTI system. We developed a real-time CUDA implementation of 3D reconstruction similar to KinectFusion [40]. First, the algorithm integrated depth images into a TSDF Volume [18]. Next, the 3D mesh was extracted from the TSDF Volume using Marching Cubes [59]. Then, the algorithm projected color images on the 3D mesh for colorization.

The resolution of TSDF volume was $256 \times 256 \times 256$ voxels. In the TSDF processing, we used a weighted average where $W = \frac{1}{Dist}$ on different cameras to minimize the error. In the colorization, we upsampled each triangle to four quartered parts to sample more colors (Figure 5). Because the users are more sensitive to the texture but not the shape [?].

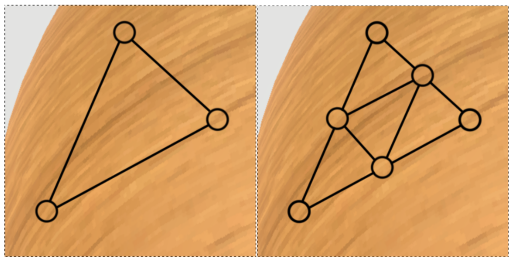

Figure 5: Left: Mesh without Supersampling; Right: Mesh with Supersampling.

We modified the TSDF algorithm to merge 3D meshes from the two capture sites. Figure 6 shows the weighted combination of the two profile from both sites. The merging rule is:

$$V_z = \min\{\frac{\sum_{local} W_{i,z} S_{i,z}}{\sum_{local} W_{i,z}}, \frac{\sum_{remote} W_{j,z} S_{j,z}}{\sum_{remote} W_{j,z}}\} \quad (1)$$

$$W_{i,z} = \frac{1}{dist(i,z)} \quad (2)$$

where $S_{i,z}$ is the Signed Distance Function (SDF) value of the zth volume from the ith camera. $dist(i,z)$ is the distance from the ith camera to the zth volume. V_z is the merged SDF value of zth volume after the combination.

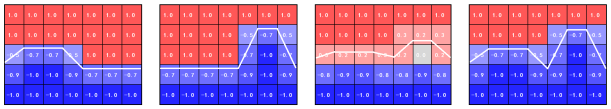

Figure 6: Left to right: 1)SDF value from camera 1; 2)Merged SDF value from camera 2; 3)SDF value by simple average; 4)Merged SDF value from our project.

End-to-end Delay

The lowest end-to-end delay of our system was 52 (± xx) ms. The frame rate of 3D reconstruction was 30 FPS. It was controlled by the frequency of the synchronized depth cameras. In average, a frame (33 ms) consisted of 19 ms processing and 14 ms idling. The remote images had one frame of latency. So the end-to-end delay was about $33 + 19 = 52$ ms. For artificial delay, we buffered remote data for frames. Figure 7 shows the pipeline in details. The rendering was independent to the reconstruction pipeline. The frame rate of rendering reached 90 FPS so that the users do not feel dizzy.

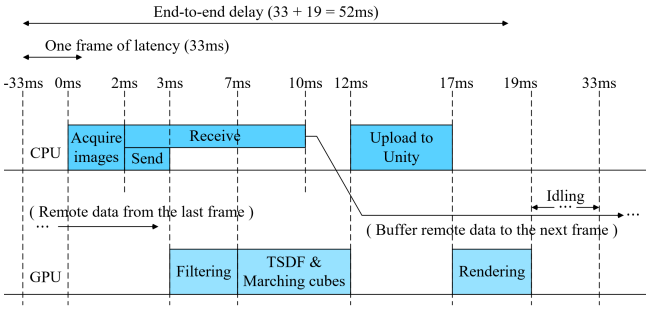

Figure 7

4 USER EXPERIMENT

[Yu] The goal of the experiment is ...

The experiment has two parts. Part A is a chess game with different conditions of cues. Results support the idea of classification in our framework. Part B is the Rock-Paper-Scissors game. It requires a low delay. We present a simple assistant design to help synchronization.

The experiment is also an example. It illustrates how to measure the noticeable delay and the acceptable delay for a specific application.

Part A: Playing Chess

Part A was to study the variety of delay perception in different synchronization level. The task was a chess game between pairs of participants in two rooms, with two conditions of cues: audiovisual mode and visual only mode.

Experimental design. We used a within-subjects experimental design. Each pair of participants played chess in two sessions of Communication Channel (CC): *audiovisual CC* and *visual only CC*, which correspond to the 2nd and 3rd synchronization level. The CC conditions were assigned to participant pairs in a Latin square design. Delay was another within-subjects factor with five conditions (50, 150, 250, 450, 750 ms in *audiovisual CC*; 150, 450, 1050, 1550, 2050 ms in *visual only CC*). In each session, we tested the five delay conditions in five trials. The delay conditions were assigned in a random order. In particular, we encouraged participants to chat in the *audiovisual CC*.

Task. In each trial, two distributed participants played chess "face-to-face" for three minutes. We adopted *Reversi* as the task. *Reversi* is simple enough that the participants can learn it in a short time. The game involves frequent interaction: when capturing, a participant should ask his partner to remove the captured chess. The participants have enough chances to perceive a noticeable delay.

In the physical world, each player interacted with a chessboard and chess pieces on his own side. The 3DTI system fused the two physical scenes into a virtual space. In the virtual space, each player could see not only his chess pieces but also his partner' s chess pieces.

Part B: Rock-Paper-Scissors

Part B was to evaluate the impact of delay in a high delay requirement situation. We used the Rock-Paper-Scissors game. We designed a synchronized audio cue to help users to synchronize with each other. The study assesses its effect on user experience.

Experimental design. This part was also a within-subjects design. Each pair played the Rock-Paper-Scissors game in two sessions: with and without the synchronized audio cue. We applied Latin square to the two sessions. In each session, we tested the delay of 50, 83, 117 and 150ms in four trials. The order of delay conditions was random. We also adopted Latin square on part A and part B.

Task. In each trial, a pair continuously drew the Rock-Paper-Scissors gestures until one of them won for ten times. There were two conditions to test: with and without the synchronized visual cue. In an actual network, it is possible to synchronize the time of two systems with almost zero milliseconds apart (the NTP protocol [?]).

Our 3DTI system provided zero-delay audio cues for both users to help them gesture exactly at the same time. The cue was an audio source of four seconds, with "tick" sounds at the 2nd, 3rd second and a "tack" sound at the 4th second. We told participants to gesture when their hear the "tack" sound.

Participants

We advertised our experiment on social media. Sixteen pairs of participants took part in our experiment (32 in total, xx females). They all came from the campus, aged from xx to xx. Participants were paid 150 yuan for the 90 minutes long study. The ten participants with the most conversation turn received extra 50 yuan.

Previous works have pointed out that the individual user differences affect study results of delay perception [?]. In our experiment, we control the source of participants carefully:

- *Relationship*: Each pair of participants are familiar with each other (friends, classmates or partners). This setting is to improve the conversation quality.
- *First language*: All the participants are native Chinese speakers. Chinese conversations are a little bit harder to predict compared to English conversations [?], which may lead to a larger noticeable delay (about xx ms). （没有找到相关文献，而且感觉中文和英文应该差不多，都是主谓宾。）
- *Experience in DIME*: Our participants have relatively high education levels. According to the self-report questionnaire, they are quite familiar with audiovisual multimedia (xx points in average, 5 for experts) and AR/VR (xx points in average).

Thus, our study results are rigorous but relatively low in the external validity. We recommend a larger amount of participants if the readers need a more general result.

Procedure

Before the experiment, we invited the participant pair to a room and explained our study. We explained the rule of *Reversi* and *Rock-Paper-Scissors*. Next, the pair had ten minutes to experience the physical interaction of these two games. We asked the participants to remember the feeling of physical interaction and regard it as a zero-delay experience. Then, we introduced our experimental procedure to the participants.

Part A had $2sessions \times 5trials = 10trials$, which lasted for 40 minutes. Part B had $2sessions \times 4trials = 8trials$ (20 minutes). In each trial, the participants experience the remote VR game. After each trial, participants filled in a short survey and rested for one minute. The questions in the survey are shown in Table 2.

After each session, participants had a five minutes break. We conducted brief interviews with some subjective questions as followed:

- How do you notice the delay? What are the cues?
- What makes you annoyed in the task?
- Any other comments?

In particular, we explained to participants about the concept of network delay. In our system, there was a local end-to-end delay of about 20 ms, which is slightly noticeable for

Label	Question	Scale
quality	How do you feel during the experiment?	Bad <–> Excellent
noticeability	Can you perceive the delay in the connection?	Very much <–> Not at all
tolerance	To what extent where you annoyed by the delay?	Severe annoyance <–> No annoyance

Table 2: Questions and scale.

the users. We told the participants that we do not consider the local delay in the questionnaire. Instead, we evaluate the network delay, which is the time interval between your partner acts and you see. A participant can perceive the network delay by judging if his partner is slow in speaking and gesturing.

Results

[NOTE from zsyzgu] I have several ideas about the result:

- In 3DTI, the users are much more tolerable than in audiovisual DIME. The immersive environment makes users focus on the interaction itself.
- A simple synchronized audio cue can help synchronization in tasks with high network requirement.
- Conversation is a stronger cue compared with visual feedback.
- Delay perception of 50 ms and 150 ms in audiovisual model have no significant difference.
- The individual user differences are larger in 3DTI.

Delay Effects. [TODO]

Figure ? shows the results for the three questionnaire items. We tried to find out when participants started to notice the delay in audio-video and video-only tasks, so we performed a pairwise comparison between the minimum delay and other delays in each mode.

In audio-video mode, the analysis shows that network quality is not significantly different($p = 0.1914$) between 50ms and 150ms, then becomes significantly different($p = 0.042$) at 450ms. For noticeability, the results follow a similar pattern. Noticeability is not significantly different($p = 0.1592$) between 50ms and 150ms, then becomes significantly different($p = 0.0061$) at 450ms. For annoyance, the result is still not significant($p = 0.1012$) at 450ms and becomes significantly($p = 0.0058$) until 750ms.

In video-only mode, the analysis shows that network quality is not significantly different($p = 0.1546$) between 150ms and 450ms, then has a significant difference($p = 0.0079$) at 1050ms. For noticeability($p = 0.0114$) and annoyance($p = 0.0326$), the result is already significant at 450ms.

In Part B, when participants can hear synchronized audio, they gave three items significantly($p < 0.05$) lower score at 117ms, but not significant($p > 0.2$) at 87ms. Without the synchronized audio, noticeability score has significant($p = 0.0174$) drop at 87ms, but for network quality($p = 0.0451$) and annoyance($p = 0.0146$), the first significant drop shows at 117ms.

In Part A, the analysis revealed that the influence of delay on the quality question was statistically significant($p = 0.0069$) for audio-visual mode. Influence of delay on noticeability was statistically significant($p = 0.0002$) for audio-visual mode. Influence of delay on annoyance was statistically significant($p = 0.0075$) for audio-visual mode. For video-only mode, the influence of delay on these items are all statistically significant($p < 0.0001$). We use t-test to analyze if the data is significantly different in the two modes. Since only 150ms and 450ms delay levels are contained in both two modes, we only analyze these two specific delays. The analysis revealed that the influence of delay between audio-visual mo de and video-only mode was not significant($p = ?$) when delay was 150ms, and was also not significant($p = ?$) when delay was 450ms(why ???).

In Part B, the analysis revealed that the influence of delay on the quality question was statistically significant($p = 0.0057$) with the synchronized audio cue but was not significant($p = 0.1551$ why???) without the synchronized audio cue. Influence of delay on noticeability was statistically significant in both modes($p = 0.0047$ with the synchronized audio cue and $p = 0.0159$ without). Influence of delay on annoyance was statistically significant($p = 0.0010$) with the synchronized audio cue but was nearly identical($p = 0.0428$) without the synchronized audio cue.

Attention: Analysis from only 6 groups of data.

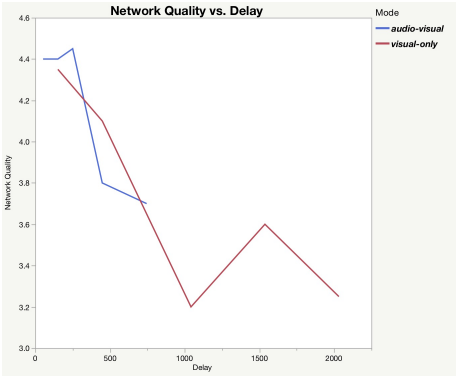

Figure 8: Effects of delay on quality score in part A(Playing Chess).

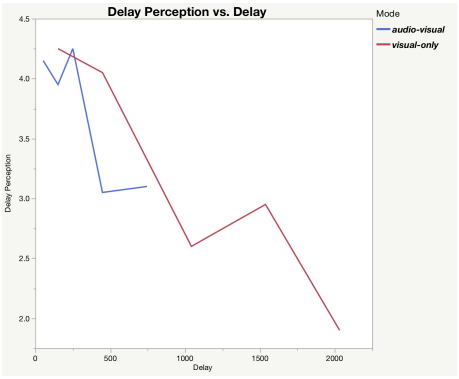

Figure 9: Effects of delay on noticeability score in part A(Playing Chess).

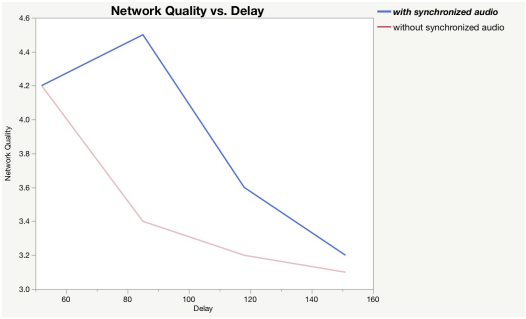

Figure 10: Effects of delay on quality score in part B(Rock-Paper-Scissors).

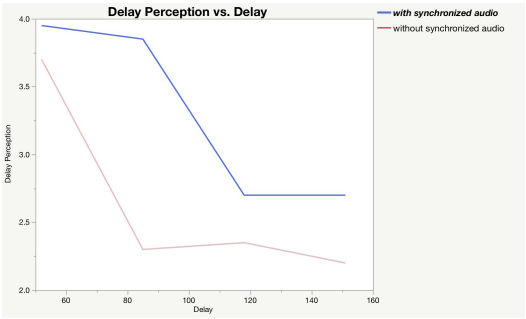

Figure 11: Effects of delay on noticeability score in part B(Rock-Paper-Scissors).

Data Analysis. [TODO]

Interview

[NOTE from zzy]

[Natural Interaction due to Immersion]

Difference	0.30000	t Radio	0.582772
Std Err Dif	0.5148	DF	15.92888
Upper CL Dif	1.3917	Prob > |t|	0.5682
Lower CL Dif	-0.7917	Prob > t	0.2841
Confidence	0.95	Prob < t	0.7159

Table 3: Oneway analysis of noticeability score by mode(Delay = 150ms)

Difference	1.0000	t Radio	1.678363
Std Err Dif	0.5958	DF	14.9258
Upper CL Dif	2.2705	Prob > |t|	0.1141
Lower CL Dif	-0.2705	Prob > t	0.0570
Confidence	0.95	Prob < t	0.9430

Table 4: Oneway analysis of noticeability score by mode(Delay = 450ms)

Our advanced 3DTI system with reconstruction and rendering in full 3D provides participants with feelings of immersion and co-presence, which leads to natural interaction between the two remote players.

By observing during the game, it is apparently noticed that most participants show body language to help with communication even during the games where they are permitted to speak to each other(e.g. leaning forward to attract the partner's attention before speaking; shrugging and spread hands wide to express confusion.)

Interestingly, at the beginning of the game, it is very easy for the participants to fall into confusion of the real and virtual objects. Even having been informed of the virtuality of everything from the other end, some participants still try to move the other's chesses. "I know there is no use in moving the black chesses, but the behavior is just like a kind of instinctive reaction, because, you know, the chesses are just being there, almost physically, which seems to be of no difference with the chess in my hand."

[Extra cues being found]

Most participants notice the delay when it is significantly high and several cues are most frequently found. In the Reversi, the expectation of instantly removing the captured chesses is a very important factor in noticing the delay. "Until she removed the captured chesses, I repeated again and again, with my fingers pointing to the target chess. She couldn't have been thinking at that time. That must have been the delay, I bet." In the Rock-Paper-Scissors, the interesting situation as follows is a noticeable cue. "It sounds ridiculous, but it does happen. I feel as if my partner throws rock after seeing I have thrown paper."

Thanks to full 3D immersion, the participants can play the game under fewer limits but with more freedom instead,

which offers chances to find extra cues.(e.g looking at the other from another angle;touching the other's hands or faces while waiting for the other's move.)"I notice that my partner is sort of obsessed with repeatedly reversing the chess in her hand until I place the chess. However, in some turns ,she didn't react instantly, for which the delay should be to blame." Some other cues are dependent on participants' relationship and experience, which should also be taken into account for QoS. "I said the funny word we invented together, and I knew she would have laughed out loud, but she didn't , until after nearly one second and a half."

[More tolerant and less noticeable with audio cues or synchronization assistance]

Feelings, about delays, of most participants agree with the randomized delays from the perspective of distribution. Being asked to compare the feelings of respective patterns, most participants show more tolerance and less noticeability with audio cues or synchronization assistance. Analysis of the interviews reveals insights as follows.

First, verbal interaction diverts attention from the delay. Instead of waiting for participants to take the next step, one can flesh up the interval with frequent talking, of which the delay is less noticeable according to some research[CITE].

Second, with frequent communication, the participants are more apt to think of the delay as consideration. It seems that without the help of verbal communication which is more informative and fast, participants are inclined to fed up with the delayed movement of the partner.

Another interesting finding is that synchronization assistance is a practical method in countermeasure to the delay. With the same groups of delays, the group in the game without the synchronization assistance is much less noticed.

5 RELATED WORK

In this section, we first review 3D tele-immersion techniques. We summarize the necessary components of current 3DTI systems to guide our implementation. Then, we have a look about existing studies on delay in 3DTI. Much more related researches were conducted in audiovisual Distributed Interactive Multimedia Environments (DIME). We discuss them later to help forming our framework.

3D Tele-immersion

Optimal techniques toward 3DTI became clear in the last decade. Basically, a 3DTI system requires three processes: reconstruction, transmission and rendering [27]. For 3D reconstruction, the volumetric algorithm has become mainstream. We applied TSDF Volume [18] and Marching Cubes [59] to fuse depth images into a polygonal mesh. We do not focus on transmission as [4, 75] did, but use 10 Gigabit optical fiber between computers instead. For rendering, we applied the head-mounted display (HTC Vive) because of its on-going growth. In this subsection, we review previous works of 3D reconstruction and rendering in details.

3D Reconstrucion. In early works, researchers used an array of cameras to capture dynamic scenes [26, 45]. For a given camera view, these systems create a polygonal model that will look correct. They do not actually construct a 3D model.

TELEPORT [29] composites video-textured surfaces within 3D geometric models. It uses only one camera. In 2002, researchers started to design virtual 3D environment with multiple cameras [32, 91]. However, their 3D reconstruction result was only point cloud. In 2008, Kurillo et al. presented a framework for remote collaboration and training of physical activities [54]. This work tried a reconstruction method with triangulation, but only reached the frame rate of about 5-7 FPS. [58] and [78] for the first time presented compelling real-time reconstruction techniques with multiple cameras. However, the lack of depth dimension indicated their modeling with only silhouette boundaries.

Researchers achieved the real-time performance of high-quality reconstruction in the last decade. In October 2011, Maimone et al. presented a 3DTI system with Kinects [61]. They developed a pixel-based mesh generation algorithm and reached a frame rate of 30 FPS. This work was followed by Beck et al.'s group-to-group telepresence system [4]. In the same month, however, Microsoft introduced KinectFusion [40] based on volumetric method. They described a novel GPU-based pipeline and achieved a better reconstruction quality. In the next year (2012), Maimone et al. also turned to volumetric methods [62] to improve the quality. A huge amount of works improved 3D reconstruction within the same framework as KinectFusion, in the region of scale [12, 71], noise reduction [46, 68, 69] and so on.

In 2016, Microsoft proposed a new pipeline named Fusion4D [21], which is highly robust to occlusions, large frame-to-frame motions, and topology changes. "The fourth dimension" is the time dimension, indicating that it leverages temporally coherence of physical scenes. In the same year, Microsoft integrated fusion4D into their 3DTI system Holoportation [73]. However, Fusion4D is extremely complex and not open-source. Even with costly devices, Holoportation has an end-to-end latency of 80ms, which can not be ignored in our study. In this paper, we apply a 3D reconstruction method similar to [62] (2012) for responsiveness.

3D Rendering. Rendering techniques in 3DTI systems can be mainly divided into three categories: light field displays, Spatially Immersive Displays (SIDs) and HMDs. The light field displays [30, 41, 44, 47] suffers from low resolution because neither computing nor rendering devices can support high-quality 4d light fields. SIDs were earlier, while HMDs are becoming popular nowadays.

Around year 2000, SIDs had become increasing significant [32]. CAVE [17] is a typical SIDs system, which consists of surround-screen projection. Users wear 3D glasses in a CAVE. Most 3DTI systems at that time applied rendering techniques similar to CAVE [5, 29, 32, 54, 91]. CAVE was design to support the one-to-many presentation. Latter researchers improved it for multi-user by using polarization or time-sharing [25, 33, 53]. Multi-user SID was used by an immersive group-to-group telepresence [4]. There is also a simplified technique called head-tracked auto-stereo display [6, 42], which allows 3D view without glasses. Some 3DTI system [61, 62, 76] used it for rendering. However, these glasses-free systems have to abandon the benefit of stereoscopy.

Recently, HMDs are becoming popular. More 3DTI systems tend to apply HMDs for 3D rendering [57, 63, 73, 88]. HMDs are basically cheaper and easier to deploy compared to SIDs. Another superiority of HMDs is their ability to support co-located collaboration [63, 73], i.e., users feel like exactly in the same place. For comparison, SIDs do not support rendering in full 3D, with which a 'window' separate users into two virtual spaces. In 2018, Microsoft proposed Remixed Reality [57]. This approach combines the benefits of augmented reality and virtual reality using 3D reconstruction and VR HMD. Users can not only see their environment but can also apply changes to it. Finally, we applied the head-mounted VR (HTC Vive) for rendering.

Delay Perception in 3DTI

User experience often relates to Quality of Service (QoS) including delay, bandwidth, jitter and packet loss [20]. Previous works have found that delay is one of the most crucial factors determining user experience in telepresence [8, 84, 86, 92]. For telephone, 150ms has been established as an industry standard for an acceptable delay [77**?**]. Also, a huge amount of related works have been conducted in audiovisual DIME.

3DTI is quite different from audiovisual DIME: first, high level of immersion offers more cues, i.e., users may be more sensitive to delay in 3DTI; second, with abundant sensory stimuli, users are more tolerant to delay [89]; third, 3DTI can support much more possible applications that we have to discuss them case by case. Thus, we have to rebuild the theoretical framework of delay perception in 3DTI.

In the area of 3DTI, most works focus on algorithm and pipeline. Negative impacts of large delay are widely reported [4, 29, 54, 61, 79]. However, only a few works were conducted to study delay perception in 3DTI [35, 95, 96]. These works do not exactly focus on delay perception. Moreover, they are limited by single scenario and immature techniques, e.g., the 2D screen was used to display 3D scenes. In this paper, we explore delay perception in a full 3D tele-immersion. We consider various scenarios and finally form a framework to understand this problem.

6 LIMITATION

1. The rendering quality of the system is not state-of-the-art.
2. The lack of eye contact.
3. The low external validity of the experiment.

7 CONCLUSION

This paragraph is for the conclusion.

ACKNOWLEDGMENTS

We thank all the volunteers, and all publications support and staff, who wrote and provided helpful comments on previous versions of this document. Authors 1, 2, and 3 gratefully acknowledge the grant from NSF (#1234–2012–ABC). *This whole paragraph is just an example.*

REFERENCES

[1] David L Allen and Herold Williams. 1996. Teleconferencing method and system for providing face-to-face, non-animated teleconference environment. US Patent 5,572,248.

[2] Ignacio Avellino, Cédric Fleury, and Michel Beaudouin-Lafon. 2015. Accuracy of deictic gestures to support telepresence on wall-sized displays. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2393–2396.

[3] Elizabeth Bates, Simona D'Amico, Thomas Jacobsen, Anna Székely, Elena Andonova, Antonella Devescovi, Dan Herron, Ching Ching Lu, Thomas Pechmann, Csaba Pléh, et al. 2003. Timed picture naming in seven languages. *Psychonomic bulletin & review* 10, 2 (2003), 344–380.

[4] Stephan Beck, Andre Kunert, Alexander Kulik, and Bernd Froehlich. 2013. Immersive group-to-group telepresence. *IEEE Transactions on Visualization and Computer Graphics* 19, 4 (2013), 616–625.

[5] Hrvoje Benko, Ricardo Jota, and Andrew Wilson. 2012. MirageTable: freehand interaction on a projected augmented reality tabletop. In *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 199–208.

[6] Hrvoje Benko, Andrew D Wilson, and Federico Zannier. 2014. Dyadic projected spatial augmented reality. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, 645–655.

[7] Susan E Brennan. 2005. How conversation is shaped by visual and spoken evidence. *Approaches to studying world-situated language use: Bridging the language-as-product and language-as-action traditions* (2005), 95–129.

[8] Kjell Brunnström, Sergio Ariel Beker, Katrien De Moor, Ann Dooms, Sebastian Egger, Marie-Neige Garcia, Tobias Hossfeld, Satu Jumisko-Pyykkö, Christian Keimel, Mohamed-Chaker Larabi, et al. 2013. Qualinet white paper on definitions of quality of experience. (2013).

[9] RECOMMENDATION ITU-R BT. 2002. Methodology for the subjective assessment of the quality of television pictures. (2002).

[10] Alexander Carôt, Pedro Rebelo, and Alain Renaud. 2007. Networked music performance: State of the art. In *Audio engineering society conference: 30th international conference: intelligent audio environments*. Audio Engineering Society.

[11] Alexander Carôt and Christian Werner. 2007. Network music performance-problems, approaches and perspectives. In *Proceedings*

of the "Music in the Global Village"-Conference, Budapest, Hungary, Vol. 162. 23–10.

[12] Jiawen Chen, Dennis Bautembach, and Shahram Izadi. 2013. Scalable real-time volumetric surface reconstruction. *ACM Transactions on Graphics (ToG)* 32, 4 (2013), 113.

[13] Jessie YC Chen and Jennifer E Thropp. 2007. Review of low frame rate effects on human performance. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 37, 6 (2007), 1063–1076.

[14] Herbert H Clark. 1981. Definite reference and mutual knowledge. *Elements of discourse understanding* (1981).

[15] Herbert H Clark and Deanna Wilkes-Gibbs. 1990. Referring as a collaborative process. *Intentions in communication* (1990), 463–493.

[16] Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. 2015. High-quality streamable free-viewpoint video. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 69.

[17] Carolina Cruz-Neira, Daniel J Sandin, and Thomas A DeFanti. 1993. Surround-screen projection-based virtual reality: the design and implementation of the CAVE. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. ACM, 135–142.

[18] Brian Curless and Marc Levoy. 1996. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. ACM, 303–312.

[19] Ricardo L de Queiroz and Philip A Chou. 2016. Compression of 3d point clouds using a region-adaptive hierarchical transform. *IEEE Transactions on Image Processing* 25, 8 (2016), 3947–3956.

[20] Angus Donovan, Leila Alem, Weidong Huang, Ren Liu, and Mark Hedley. 2014. Understanding How Network Performance Affects User Experience of Remote Guidance. In *CYTED-RITOS International Workshop on Groupware*. Springer, 1–12.

[21] Mingsong Dou, Sameh Khamis, Yury Degtyarev, Philip Davidson, Sean Ryan Fanello, Adarsh Kowdle, Sergio Orts Escolano, Christoph Rhemann, David Kim, Jonathan Taylor, et al. 2016. Fusion4d: Real-time performance capture of challenging scenes. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 114.

[22] Thomas Enderes, Swee Chern Khoo, Clare A Somerville, and Kostas Samaras. 2002. Impact of statistical multiplexing on voice quality in cellular networks. *Mobile networks and applications* 7, 2 (2002), 153–161.

[23] Mica R Endsley. 2017. Toward a theory of situation awareness in dynamic systems. In *Situational Awareness*. Routledge, 9–42.

[24] Mica R Endsley and Daniel J Garland. 2000. *Situation awareness analysis and measurement*. CRC Press.

[25] Bernd Fröhlich, Jan Hochstrate, Jörg Hoffmann, Karsten Klüger, Roland Blach, Matthias Bues, and Oliver Stefani. 2005. Implementing multi-viewer stereo displays. (2005).

[26] Henry Fuchs, Gary Bishop, Kevin Arthur, Leonard McMillan, Ruzena Bajcsy, Sang Lee, Hany Farid, and Takeo Kanade. 1994. Virtual space teleconferencing using a sea of cameras. In *Proc. First International Conference on Medical Robotics and Computer Assisted Surgery*, Vol. 26.

[27] Henry Fuchs, Andrei State, and Jean-Charles Bazin. 2014. Immersive 3d telepresence. *Computer* 47, 7 (2014), 46–52.

[28] Darren Gergle, Robert E Kraut, and Susan R Fussell. 2006. The impact of delayed visual feedback on collaborative performance. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, 1303–1312.

[29] Simon J Gibbs, Constantin Arapis, and Christian J Breiteneder. 1999. TELEPORT–Towards immersive copresence. *Multimedia Systems* 7, 3 (1999), 214–221.

[30] Daniel Gotsch, Xujing Zhang, Timothy Merritt, and Roel Vertegaal. 2018. TeleHuman2: A Cylindrical Light Field Teleconferencing System for Life-size 3D Human Telepresence. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 522.

[31] Zenzi M Griffin and Kathryn Bock. 2000. What the eyes say about speaking. *Psychological science* 11, 4 (2000), 274–279.

[32] Markus Gross, Stephan Würmlin, Martin Naef, Edouard Lamboray, Christian Spagno, Andreas Kunz, Esther Koller-Meier, Tomas Svoboda, Luc Van Gool, Silke Lang, et al. 2003. blue-c: a spatially immersive display and 3D video portal for telepresence. In *ACM Transactions on Graphics (TOG)*, Vol. 22. ACM, 819–827.

[33] Dongdong Guan, Chenglei Yang, Weisi Sun, Yuan Wei, Wei Gai, Yulong Bian, Juan Liu, Qianhui Sun, Siwei Zhao, and Xiangxu Meng. 2018. Two Kinds of Novel Multi-user Immersive Display Systems. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 599.

[34] Yousuke Hashimoto and Yutaka Ishibashi. 2006. Influences of network latency on interactivity in networked rock-paper-scissors. In *Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games*. ACM, 23.

[35] Zixia Huang, Ahsan Arefin, Pooja Agarwal, Klara Nahrstedt, and Wanmin Wu. 2012. Towards the understanding of human perceptual quality in tele-immersive shared activity. In *Proceedings of the 3rd Multimedia Systems Conference*. ACM, 29–34.

[36] Peter Indefrey and Willem JM Levelt. 2004. The spatial and temporal signatures of word production components. *Cognition* 92, 1-2 (2004), 101–144.

[37] Ellen A Isaacs and John C Tang. 1994. What video can and cannot do for collaboration: a case study. *Multimedia systems* 2, 2 (1994), 63–73.

[38] T ITU. 2003. Recommendation G. 107 The E-model, a computational model for use in transmission planning. (2003).

[39] P ITU-T RECOMMENDATION. 1998. Subjective audiovisual quality assessment methods for multimedia applications. (1998).

[40] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. 2011. KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*. ACM, 559–568.

[41] Andrew Jones, Ian McDowall, Hideshi Yamada, Mark Bolas, and Paul Debevec. 2007. Rendering for an interactive 360 light field display. *ACM Transactions on Graphics (TOG)* 26, 3 (2007), 40.

[42] Brett Jones, Rajinder Sodhi, Michael Murdock, Ravish Mehra, Hrvoje Benko, Andrew Wilson, Eyal Ofek, Blair MacIntyre, Nikunj Raghuvanshi, and Lior Shapira. 2014. RoomAlive: magical experiences enabled by scalable, adaptive projector-camera units. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, 637–644.

[43] Norman P Jouppi, Daniel J Scales, and Wayne Roy Mack. 2001. Robotic telepresence system. US Patent 6,292,713.

[44] Joel Jurik, Andrew Jones, Mark Bolas, and Paul Debevec. 2011. Prototyping a light field display involving direct observation of a video projector array. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*. IEEE, 15–20.

[45] Takeo Kanade, Peter Rander, and PJ Narayanan. 1997. Virtualized reality: Constructing virtual worlds from real scenes. *IEEE multimedia* 4, 1 (1997), 34–47.

[46] Kourosh Khoshelham and Sander Oude Elberink. 2012. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors* 12, 2 (2012), 1437–1454.

[47] Kibum Kim, John Bolton, Audrey Girouard, Jeremy Cooperstock, and Roel Vertegaal. 2012. TeleHuman: effects of 3d perspective on gaze

and pose estimation with a life-size cylindrical telepresence pod. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2531–2540.

[48] Itoh K Kitawaki N. 1991. Pure Delay Effect on Speech Quality in Telecommunications. *IEEE J. Sel. Areas Comm*, 586–593.

[49] Leonard Kleinrock. 1992. The latency/bandwidth tradeoff in gigabit networks. *IEEE Communications Magazine* 30, 4 (1992), 36–40.

[50] Robert M Krauss and Peter D Bricker. 1967. Effects of transmission delay and access delay on the efficiency of verbal communication. *The Journal of the Acoustical Society of America* 41, 2 (1967), 286–292.

[51] Robert E Kraut, Susan R Fussell, and Jane Siegel. 2003. Visual information as a conversational resource in collaborative physical tasks. *Human–Computer Interaction* 18, 1-2 (2003), 13–49.

[52] Robert E Kraut, Darren Gergle, and Susan R Fussell. 2002. The use of visual information in shared visual spaces: Informing the development of virtual co-presence. In *Proceedings of the 2002 ACM conference on Computer supported cooperative work*. ACM, 31–40.

[53] Alexander Kulik, André Kunert, Stephan Beck, Roman Reichel, Roland Blach, Armin Zink, and Bernd Froehlich. 2011. C1x6: a stereoscopic six-user display for co-located collaboration in shared virtual environments. In *ACM Transactions on Graphics (TOG)*, Vol. 30. ACM, 188.

[54] Gregorij Kurillo, Ruzena Bajcsy, Klara Nahrsted, and Oliver Kreylos. 2008. Immersive 3d environment for remote collaboration and training of physical activities. In *Virtual Reality Conference, 2008. VR'08. IEEE*. IEEE, 269–270.

[55] Stephen C Levinson. 2016. Turn-taking in human communication–origins and implications for language processing. *Trends in cognitive sciences* 20, 1 (2016), 6–14.

[56] Stephen C Levinson and Francisco Torreira. 2015. Timing in turn-taking and its implications for processing models of language. *Frontiers in psychology* 6 (2015), 731.

[57] David Lindlbauer and Andy D Wilson. 2018. Remixed Reality: Manipulating Space and Time in Augmented Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 129.

[58] Charles Loop, Cha Zhang, and Zhengyou Zhang. 2013. Real-time high-resolution sparse voxelization with application to image-based modeling. In *Proceedings of the 5th High-Performance Graphics Conference*. ACM, 73–79.

[59] William E Lorensen and Harvey E Cline. 1987. Marching cubes: A high resolution 3D surface construction algorithm. In *ACM siggraph computer graphics*, Vol. 21. ACM, 163–169.

[60] Paul Luff and Christian Heath. 1998. Mobility in collaboration. In *Proceedings of the 1998 ACM conference on Computer supported cooperative work*. ACM, 305–314.

[61] Andrew Maimone and Henry Fuchs. 2011. Encumbrance-free telepresence system with real-time 3D capture and display using commodity depth cameras. In *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*. IEEE, 137–146.

[62] Andrew Maimone and Henry Fuchs. 2012. Real-time volumetric 3D capture of room-sized scenes for telepresence. In *3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON), 2012*. IEEE, 1–4.

[63] Andrew Maimone, Xubo Yang, Nate Dierk, Andrei State, Mingsong Dou, and Henry Fuchs. 2013. General-purpose telepresence with head-worn optical see-through displays and projector-based lighting. In *Virtual Reality (VR), 2013 IEEE*. IEEE, 23–26.

[64] Jennifer Marlow, Scott Carter, Nathaniel Good, and Jung-Wei Chen. 2016. Beyond talking heads: multimedia artifact creation, use, and sharing in distributed meetings. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*. ACM, 1703–1715.

[65] David L Mills. 1991. Internet time synchronization: the network time protocol. *IEEE Transactions on communications* 39, 10 (1991), 1482–1493.

[66] Kana Misawa and Jun Rekimoto. 2015. ChameleonMask: Embodied physical and social telepresence using human surrogates. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, 401–411.

[67] Carman Neustaedter, Gina Venolia, Jason Procyk, and Daniel Hawkins. 2016. To Beam or not to Beam: A study of remote telepresence attendance at an academic conference. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*. ACM, 418–431.

[68] Richard A Newcombe, Dieter Fox, and Steven M Seitz. 2015. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 343–352.

[69] Chuong V Nguyen, Shahram Izadi, and David Lovell. 2012. Modeling kinect sensor noise for improved 3d reconstruction and tracking. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on*. IEEE, 524–530.

[70] Jakob Nielsen. 1993. Response times: the three important limits. *Usability Engineering* (1993).

[71] Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Marc Stamminger. 2013. Real-time 3D reconstruction at scale using voxel hashing. *ACM Transactions on Graphics (ToG)* 32, 6 (2013), 169.

[72] Brid O'CONAILL. 1997. Characterizing, predicting and measuring video-mediated communication: a conversational approach. *Video mediated communication* (1997).

[73] Sergio Orts-Escolano, Christoph Rhemann, Sean Fanello, Wayne Chang, Adarsh Kowdle, Yury Degtyarev, David Kim, Philip L Davidson, Sameh Khamis, Mingsong Dou, et al. 2016. Holoportation: Virtual 3d teleportation in real-time. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, 741–754.

[74] Lothar Pantel and Lars C Wolf. 2002. On the impact of delay on real-time multiplayer games. In *Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video*. ACM, 23–29.

[75] Fabrizio Pece, Jan Kautz, and Tim Weyrich. 2011. Adapting standard video codecs for depth streaming. In *Proceedings of the 17th Eurographics conference on Virtual Environments & Third Joint Virtual Reality*. Eurographics Association, 59–66.

[76] Tomislav Pejsa, Julian Kantor, Hrvoje Benko, Eyal Ofek, and Andrew Wilson. 2016. Room2room: Enabling life-size telepresence in a projected augmented reality environment. In *Proceedings of the 19th ACM conference on computer-supported cooperative work & social computing*. ACM, 1716–1725.

[77] Alan Percy. 1999. Understanding latency in IP telephony. *Brooktrout Technology, Needham, MA* (1999).

[78] Benjamin Petit, Jean-Denis Lesage, Clément Menier, Jérémie Allard, Jean-Sébastien Franco, Bruno Raffin, Edmond Boyer, and François Faure. 2010. Multicamera real-time 3d modeling for telepresence and remote collaboration. *International journal of digital multimedia broadcasting* 2010 (2010).

[79] Suraj Raghuraman and Balakrishnan Prabhakaran. 2015. Distortion score based pose selection for 3D tele-immersion. In *Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology*. ACM, 227–236.

[80] ITUT Rec. 2006. P. 800.1, Mean opinion score (MOS) terminology. *International Telecommunication Union, Geneva* (2006).

[81] G Recommendation. 2003. 114-One-way transmission time ITU.

[82] Harvey Sacks, Emanuel A Schegloff, and Gail Jefferson. 1978. A simplest systematics for the organization of turn taking for conversation. In *Studies in the organization of conversational interaction*. Elsevier, 7–55.

[83] Christian Schaefer, Thomas Enderes, Hartmut Ritter, and Marina Zitterbart. 2002. Subjective quality assessment for multiplayer real-time games. In *Proceedings of the 1st workshop on Network and system support for games*. ACM, 74–78.

[84] Marwin Schmitt, Simon Gunkel, Pablo Cesar, and Dick Bulterman. 2014. Asymmetric delay in video-mediated group discussions. In *Quality of Multimedia Experience (QoMEX), 2014 Sixth International Workshop on*. IEEE, 19–24.

[85] Marwin Schmitt, Simon Gunkel, Pablo Cesar, and Dick Bulterman. 2014. The influence of interactivity patterns on the Quality of Experience in multi-party video-mediated conversations under symmetric delay conditions. In *Proceedings of the 3rd International Workshop on Socially-aware Multimedia*. ACM, 13–16.

[86] Marwin Schmitt, Simon Gunkel, Pablo Cesar, and Peter Hughes. 2013. A QoE testbed for socially-aware video-mediated group communication. In *Proceedings of the 2nd international workshop on Socially-aware multimedia*. ACM, 37–42.

[87] Nathan Schuett. 2002. The effects of latency on ensemble performance. *Bachelor Thesis, CCRMA Department of Music, Stanford University* (2002).

[88] Harrison Jesse Smith and Michael Neff. 2018. Communication Behavior in Embodied Virtual Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 289.

[89] Jennifer Tam, Elizbeth Carter, Sara Kiesler, and Jessica Hodgins. 2012. Video increases the perception of naturalness during remote interactions with latency. In *CHI'12 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2045–2050.

[90] John C Tang and Ellen Isaacs. 1992. Why do users like video? *Computer Supported Cooperative Work (CSCW)* 1, 3 (1992), 163–196.

[91] Herman Towles, Wei-Chao Chen, Ruigang Yang, Sang-Uok Kum, Henry Fuchs Nikhil Kelshikar, Jane Mulligan, Kostas Daniilidis, Henry Fuchs, Carolina Chapel Hill, Nikhil Kelshikar Jane Mulligan, et al. 2002. 3d tele-collaboration over internet2. In *In: International Workshop on Immersive Telepresence, Juan Les Pins*. Citeseer.

[92] Andreas Vogel, Brigitte Kerherve, Gregor von Bochmann, and Jan Gecsei. 1995. Distributed multimedia and QoS: A survey. *IEEE multimedia* 2, 2 (1995), 10–19.

[93] Jian Wang, Fuzheng Yang, Zhiqing Xie, and Shuai Wan. 2010. Evaluation on perceptual audiovisual delay using average talkspurts and delay. In *Image and Signal Processing (CISP), 2010 3rd International Congress on*, Vol. 1. IEEE, 125–128.

[94] Steve Whittaker. 2003. Things to talk about when talking about things. *Human–Computer Interaction* 18, 1-2 (2003), 149–170.

[95] Wanmin Wu, Ahsan Arefin, Zixia Huang, Pooja Agarwal, Shu Shi, Raoul Rivas, and Klara Nahrstedt. 2010. " I'm the Jedi!"-A Case Study of User Experience in 3D Tele-immersive Gaming. In *Multimedia (ISM), 2010 IEEE International Symposium on*. IEEE, 220–227.

[96] Wanmin Wu, Ahsan Arefin, Raoul Rivas, Klara Nahrstedt, Renata Sheppard, and Zhenyu Yang. 2009. Quality of experience in distributed interactive multimedia environments: toward a theoretical framework. In *Proceedings of the 17th ACM international conference on Multimedia*. ACM, 481–490.