A Sharing Method of 3D Physical Objects with Interactive Manipulations in Real-time over the Internet by Web Real-time Communication

○ Daichi ISHIKAWA[*1], Tomohiro FUKUDA[*2] and Nobuyoshi YABUKI[*3]

*1 Master's Course Student, Graduate School of Engineering, Osaka University, B.Eng.

*2 Associate Professor, Graduate School of Engineering, Osaka University, Ph.D.

*3 Professor, Graduate School of Engineering, Osaka University, Ph.D.

Keywords: Internet; Web Real-Time Communication (WebRTC); 3D remote sharing; interactivity; point cloud data.

1. Introduction

In the architectural fields, the physical environment is often shared such as a design study of three-dimensional (3D) space using architectural models and exchanging information in construction management among construction sites and construction offices. Sharing the physical environment among remote areas is accomplished by web meetings using web cameras and flat computer displays instead of face-to-face meetings. Also, we are forced to use web meetings when pandemic or natural disasters, thus, developing remote collaboration tools is meaningful. In conventional web meetings, however, a sense of presence is lost compared with face-to-face meetings[1)]. Moreover, Sharing 3D information that is required in the architectural fields is difficult.

Telepresence which is a concept and technology to allow people to feel the presence of something in a remote area[2)], is enabled to solve the challenges of the conventional web meetings. Mixed Reality (MR)[3)] that integrates a virtual environment with the physical environment is useful for telepresence. Kantonen et al. developed a telepresence system using MR for sharing 3D virtual models created by Building Information Modeling (BIM) software[4)]. In this system, however, users could not share unexpected changes of 3D physical objects with other users in a remote area when they tried to share 3D physical objects as well as users required time to create 3D virtual models.

To solve the problem, Ishikawa et al.[5)] presented a 3D remote sharing method that could share point cloud objects created by point cloud data which was captured by an RGB-D camera in real-time, segmented by a server PC, and transferred to a remote area, on MR-Head Mounted Display (HMD). In this method, MR-HMD users could see the result of other user's manipulation of each point cloud object in the same place in real-time, due to sharing the MR coordinate system among all MR-HMDs and high-speed point cloud segmentation. However, all users had to be under the same Local Area Network (LAN) environment, that's why the usage area for the users was limited. To realize a remote collaboration without the physical limitation, a method to transfer the segmented point cloud data to a remote area in real-time over the Internet is demanded. Blanco-Novoa et al.[6)] accomplished Internet communication in real-time between an MR-Head Mounted Display (HMD) and Internet of Things (IoT) devices, but the size of transferred data was not large such as point cloud data.

To realize the data communication in real-time, Web Real-time Communication (WebRTC) is useful. WebRTC is a standard technology to enable real-time communication among web browsers in a peer-to-peer (P2P) fashion after a signaling process that exchanges peer information such as IP addresses and port numbers for data communication among each device[7)]. Compared with WebSocket that is a server-client fashion for data communication, WebRTC is faster[8)] and can communicate to the external network even if Network Address Translation (NAT) and a firewall exist.

In this research, we consider availability for WebRTC as the Internet communication method for segmented point cloud data in real-time to extend the usage area of our previous research[5)]. In the proposed system, hence, the system architecture is reconsidered, especially the part of data communication from our previous research. Also, we construct a prototype system of our proposed method and evaluate the latency of communication between the devices and user experience.

2. Proposed Method

2.1. SYSTEM OVERVIEW

Figure 1 shows our proposed method. In site A, point cloud data captured by an RGB-D camera is segmented by high-speed point cloud segmentation in a server PC and classified into individual clusters.

Then, the server PC in site A communicates to the MR-HMD in site B in a P2P fashion over the Internet using WebRTC after a signaling process. In site B, an MR-HMD user

日本建築学会情報システム技術委員会
第43回情報･システム･利用･技術シンポジウム論文集, 116-119, 2020年12月, オンライン
Proceedings of the 43rd Symposium on Computer Technology of Information, Systems and Applications, AIJ, 116-119, Dec., 2020, Online

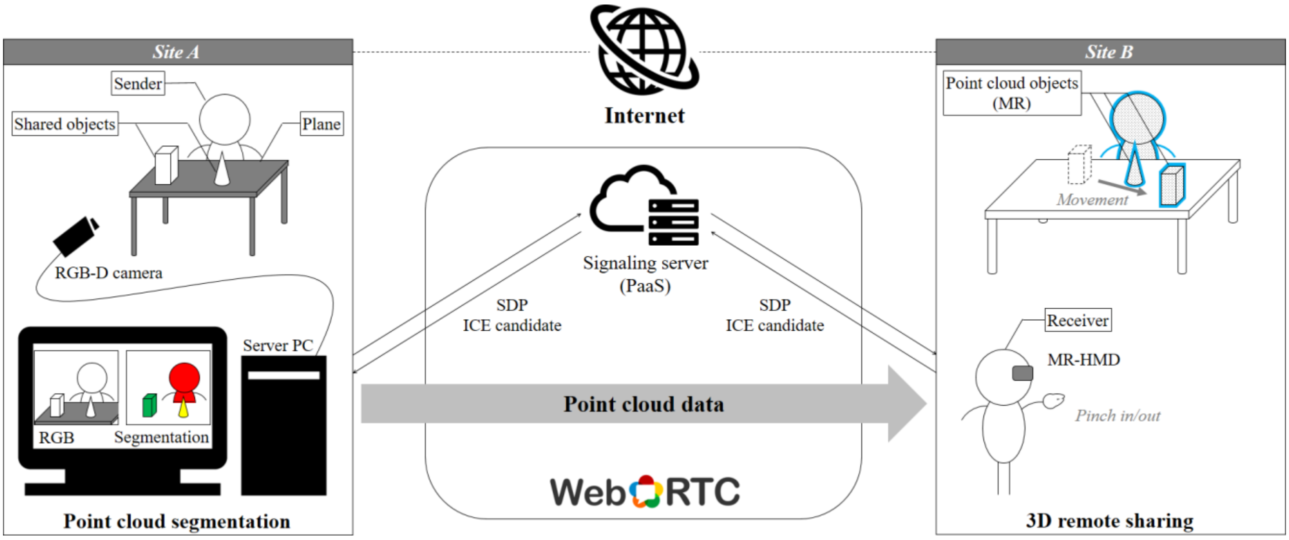

Figure 1. System overview.

can share point cloud objects with interactive manipulations. In the proposed method, sharing MR coordinates among multiple MR-HMD[5] is not applied because the data communication path is different from point cloud data communication's one and required to build it anew.

2.2. SIGNALING PROCESS

Before P2P communication using WebRTC, each peer device such as a server PC and an MR-HMD is required for a signaling process. In the proposed method, a signaling server that is required for the signaling process is built by using Platform as a Service (PaaS), which is one of the cloud services, not to need to manage the server in a local area. First, each peer device exchanges its own IP address and port numbers for data communication by Session Description Protocol (SDP). Next, each peer device exchanges Interactive Connectivity Establishment (ICE), which is the information on the available communication channel. After their process is finished, each peer device can start P2P communication.

2.3. POINT CLOUD DATA COMMUNICATION

In site A, segmented point cloud data has XYZ coordinates, RGB colors and cluster numbers. In our proposed method, the point cloud data is converted into a one-dimensional byte array to transfer it to site B (see Figure 2).

Point cloud data communication is not required a web server for their communication thanks to the P2P communication between a server PC and an MR-HMD. The point cloud data is transferred every 100 frames by the server PC, whereas the MR-HMD waits in receivable status every frame to be able to receive the transferred frame anytime in our reconsidered system architecture (see Figure 3).

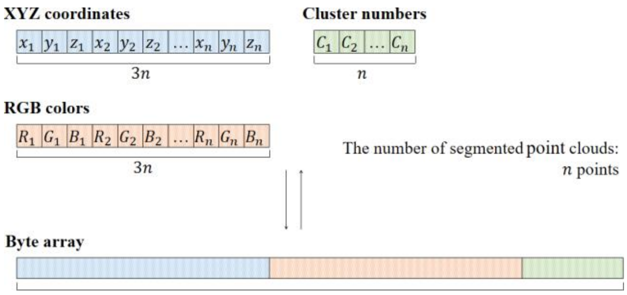

Figure 2. Conversion of point cloud data.

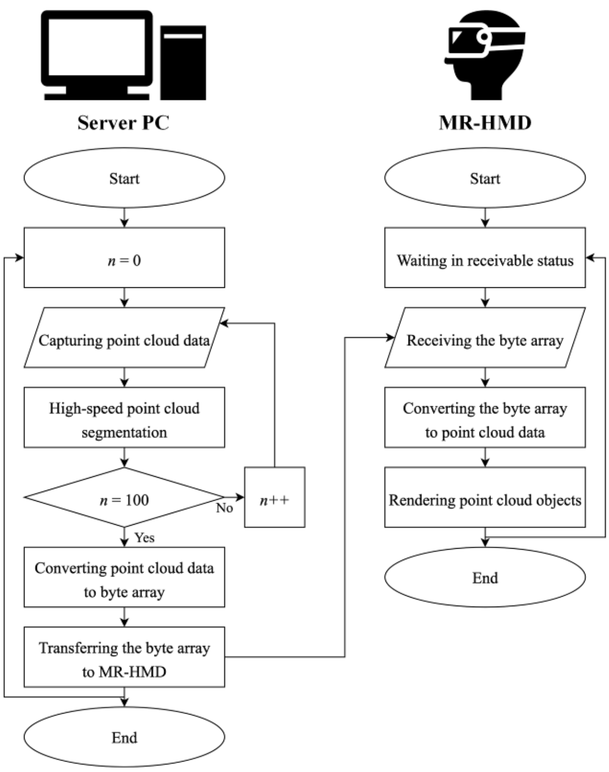

Figure 3. Point cloud data communication.

2.4. CREATING POINT CLOUD OBJECTS

In site B, received point cloud data consisted of a one-dimensional byte array by an MR-HMD is converted to XYZ coordinates, RGB colors and cluster numbers again and rendered as squares to create point cloud objects using MR in the real world. The squares are decided on their positions and colors based on XYZ coordinates and RGB colors of the point cloud data. Moreover, each point recognizes other points that have the same cluster number as the points to compose the same point cloud object.

3. Experiment and Results

3.1. PROTOTYPE SYSTEM

To evaluate the proposed system, we constructed a prototype system. In this system, We used a desktop PC running Windows 10 with Intel Core i7-7700K of CPU, 16 GB of RAM and NVIDIA Geforce GTX 1070 Ti of GPU for high-speed point cloud segmentation and point cloud data communication using WebRTC as a server PC. The desktop PC was connected wired LAN to keep the speed of transferring data but can also be used wireless LAN. An Intel RealSense Depth Camera D435i (hereinafter, RealSense), which can capture point cloud data from 0.1 m of the body to 10.0 m, a 69.4° horizontal and 42.5° vertical field of view, was used as an RGB-D camera. Furthermore, a Microsoft HoloLens (1st gen) (hereinafter, HoloLens) with Intel 32-bit of CPU and 2.0 GB of RAM was also used as an MR-HMD. The MR-HMD was connected to the access point (AP) by wireless LAN. Also, audio functions (microphones and speakers) were technically possible but not implemented to avoid decreasing the transfer speed of point cloud data at this time.

We used Microsoft Visual Studio for the integrated development environment (IDE) and Unity for the game engine to develop the prototype system. Moreover, a signaling server was run by a program made by Node.js that is a JavaScript runtime for a server-side, on Heroku which is one of the PaaS. Internet communication using WebRTC was realized by Mixed Reality-WebRTC which is a library to use WebRTC on MR applications.

3.2. EXPERIMENTAL ENVIRONMENT

To evaluate the prototype system, we used it with two people, one was a sender in Room A and another was a receiver in Room B (see Figure 4). In Room A, point cloud data of shared objects such as the sender and a 3D physical objects that was a windmill figurine with thin blades (W90 x D85 x H165 mm) was captured by a RealSense. In Room B, the receiver saw the shared objects in site A as point cloud objects through HoloLens. Also, AP for the HoloLens was run by tethering on a smartphone. We measured the Internet communication speed of networks connected to the server PC and HoloLens by a website[9] before the experiment (see Table 1). The communication speed was measured 5 times because it was not stable, and we decided the average as the measured value.

3.3. RESULTS

Figure 5 shows point cloud objects that the receiver saw through the HoloLens. We found out that the receiver could see the shared objects as the point cloud objects under the Internet environment using WebRTC. Moreover, the average processing speed was 6.8 fps.

Figure 6 shows the relationship between the number of communicated point clouds and the latency of the data communication between the sender and the receiver. The latency was 1−2 seconds when the number of point clouds was 5000−8000 points, that not affected for user experience.

4. Discussion

Through using the prototype system that we developed, we confirmed that segmented point cloud data could be communicated in real-time under the Internet environment by using WebRTC in a P2P fashion. Compared with our previous research[5], the physical limitation for users was overcome.

Table 1. Measured communication speed.

Peer device	Type	Speed [Mbps]	
Server PC	Wired LAN	Upload	888
		Download	798
HoloLens	Wireless LAN (Tethering)	Upload	10.3
		Download	31.6

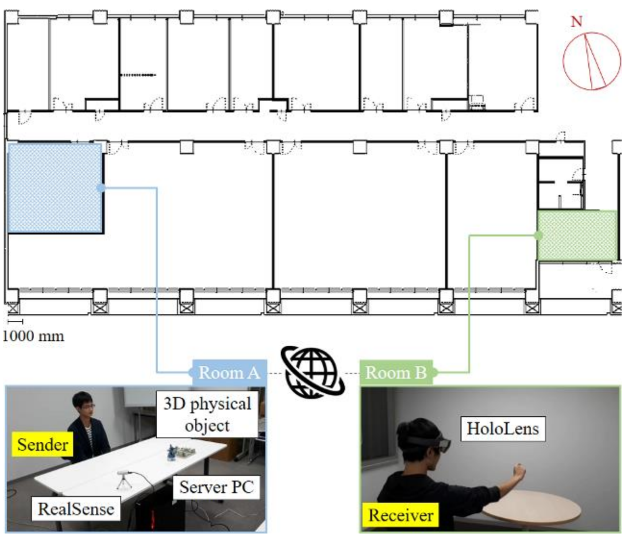

Figure 4. The position of the sender and the receiver.

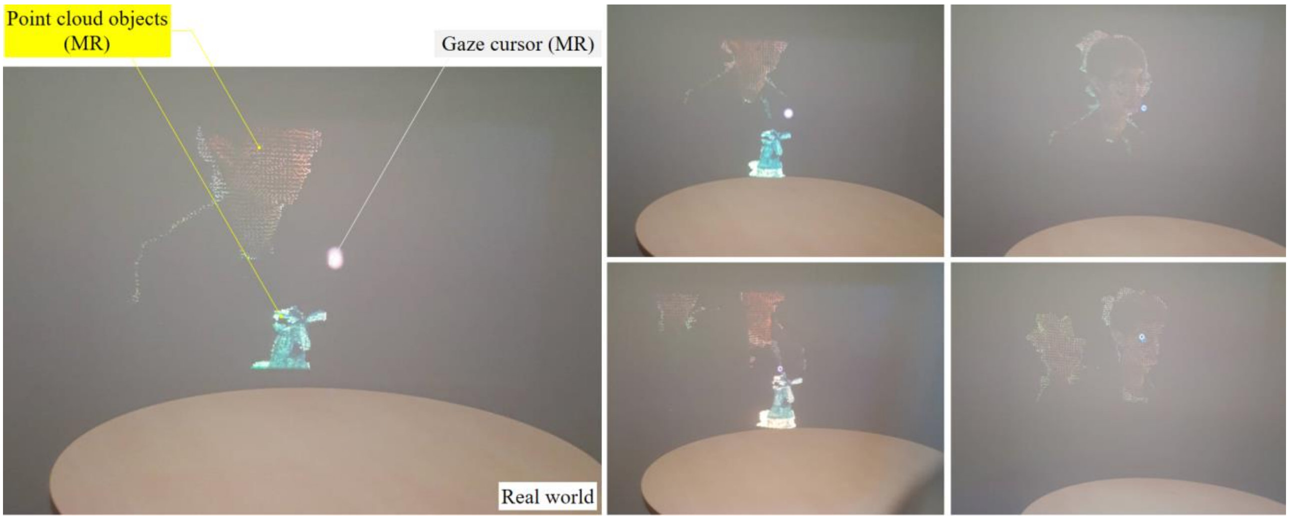

Figure 5. Pictures of point cloud objects took by a camera through HoloLens.

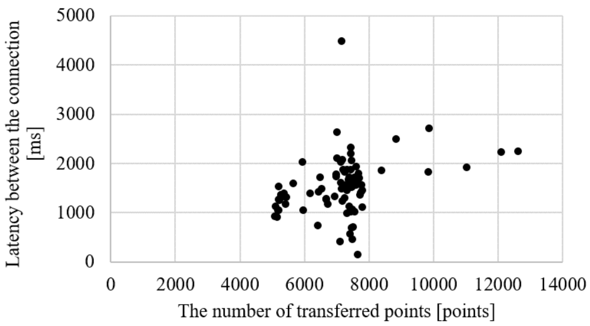

Figure 6. The relation between the number of communicated point cloud data and the latency of the data communication.

However, we found challenges. Firstly, we used just one RealSense in the proposed system, that is why point cloud data was only captured from the RealSense side and the point cloud objects only created one side. To solve this problem, we have to add RealSense to capture point cloud data from more angles and merge them in point cloud processing.

Next, we have to compact the device by using smartphones and tablets instead of an MR-HMD to be familiar with normal people.Thus, a data compression method in real-time has to be applied to our proposed system to reduce data size for the communication.

5. Conclusions and Future Works

The conclusions of this research are listed below.

- We accomplished communicating segmented point cloud data under over the Internet by using WebRTC.
- The latency of communication between the sender and the receiver was within 1−2 seconds, that not affected for user experience.

In the future, we have to add the number of RGB-D cameras to create 3D point cloud objects that have all surfaces and apply a data compression method in real-time into our proposed method to compact the device.

Acknowledgments

This research has been partly supported by the research grant of Nohmura Foundation for Membrane Structure’s Technology.

References

1) Hollan, J. and Stornetta, S.: 1992, Beyond being there, In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '92), 119–125.

2) Draper, J.V., Kaber, D.B. and Usher, J.M.: 1998, Telepresence, Human factors, 40(3), 354–375.

3) Milgram, P. and Kishino, F.: 1994, A taxonomy of mixed reality visual displays, IEICE Transactions on Information and Systems, E77-D(12), 1321–1329.

4) Kantonen, T., Woodward, C. and Katz, N.: 2010, Mixed reality in virtual world teleconferencing, 2010 IEEE Virtual Reality Conference (VR), 179–182.

5) Ishikawa, D., Fukuda, T. and Yabuki, N.: 2020, A mixed reality coordinate system for multiple HMD users manipulating real-time point cloud objects, Proceedings of the 38th eCAADe Conference, 1, 197–206.

6) Blanco-Novoa, Ó., Fraga-Lamas, P., Vilar-Montesinos, M., Fernández-Caramés, T.M.:2020, Creating the internet of augmented things: An open-source framework to make iot devices and augmented and mixed reality systems talk to each other, Sensors 2020, 20, 3328.

7) Loreto, S. and Romano, S.: 2012, Real-time communications in the web: Issues, achievements, and ongoing standardization efforts, IEEE Internet Computing, 16(5), 68–73.

8) Karla, T., Tarnawski, J.: 2019, Soft real-time communication with websocket and WebRTC protocols performance analysis for web-based control loops, 2019 24th International Conference on Methods and Models in Automation and Robotics (MMAR), 634–639.

9) Fast.com: “Fast.com: Internet Speed Test”. Available from: <https://fast.com/ja/> (accessed 27 October 2020).