2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)
November 28-30, 2022. Ginowan, Japan.

Enhanced Visual Feedback with Decoupled Viewpoint Control in Immersive Humanoid Robot Teleoperation using SLAM

Yang Chen*, Leyuan Sun*, Mehdi Benallegue, Rafael Cisneros-Limón, Rohan P. Singh, Kenji Kaneko, Arnaud Tanguy, Guillaume Caron, Kenji Suzuki, Abderrahmane Kheddar, and Fumio Kanehiro

2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids) | 979-8-3503-0979-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/HUMANOIDS53995.2022.9999740

***Abstract*— In immersive humanoid robot teleoperation, there are three main shortcomings that can alter the transparency of the visual feedback: (i) the lag between the motion of the operator's and robot's head due to network communication delays or slow robot joint motion. This latency could cause a noticeable delay in the visual feedback, which jeopardizes the embodiment quality, can cause dizziness, and affects the interactivity resulting in operator frequent motion pauses for the visual feedback to settle; (ii) the mismatch between the camera's and the headset's field-of-views (FOV), the former having generally a lower FOV; and (iii) a mismatch between human's and robot's range of motions of the neck, the latter being also generally lower. In order to leverage these drawbacks, we developed a decoupled viewpoint control solution for a humanoid platform which allows visual feedback with low-latency and artificially increases the camera's FOV range to match that of the operator's headset. Our novel solution uses SLAM technology to enhance the visual feedback from a reconstructed mesh, complementing the areas that are not covered by the visual feedback from the robot. The visual feedback is presented as a point cloud in real-time to the operator. As a result, the operator is fed with real-time vision from the robot's head orientation by observing the pose of the point cloud. Balancing this kind of awareness and immersion is important in virtual reality based teleoperation, considering the safety and robustness of the control system. An experiment shows that the effectiveness of our solution.**

***Index Terms*— VR teleoperation, humanoid, latency, immersive**

I. Introduction

Teleoperation technology is getting a renewed attention due to its potential in transferring human's intelligence and actions to a remote location, whose interest is even more relevant due to the current Covid-19 outbreak. ANA Avatar XPrize [1] is one of the robotics competitions encouraging this trend. In teleoperation, a common setup consists of an operator equipped with a master station to control, through network communication, a robot situated at a remote location. For an immersive control of the robot's head motion and receiving a visual feedback from the robot, a wearable head-mounted display (HMD) is the most common choice. One of the common issues faced in teleoperation is the lag between operator's and robot's head motion due to network communication delays or slow robot joint motion. The visual rendering of the robot's vision sensor (e.g., embedded stereo camera) is delayed. In an immersive experience, such rendering latency causes dizziness especially when the operator moves her/his head fast [2]. If the viewpoint of HMD is decoupled from the visual information or if its FOV is larger than that of the camera, then there is a blank area that is not covered by the visual feedback from the robot's camera.

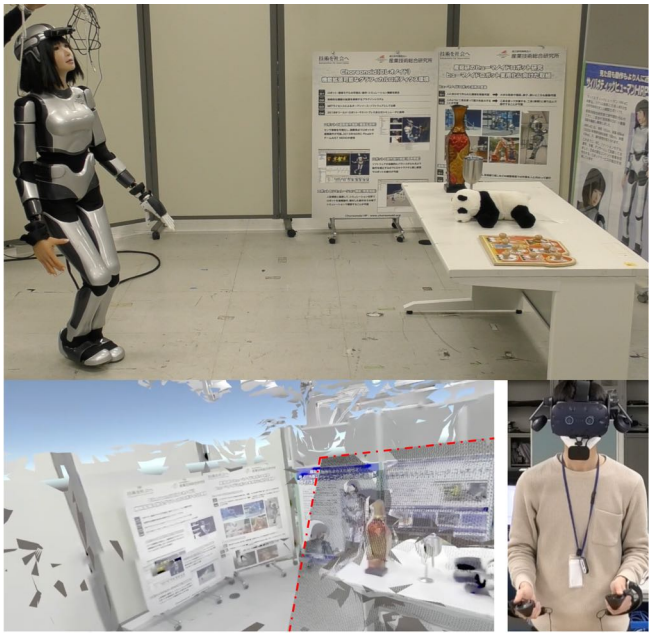

Fig. 1. Proposed tele-visualization. Top side: real scene, left-down side: virtual scene, the real-time point cloud is shown in the area inside the red dot line, the area outside denotes the constructed mesh, right-down side: operator.

Furthermore, humanoid robots often have a lower range of motion w.r.t humans, which may also limit the visual perception if the operator's master station does not constrain the operator head to match the robot's head motion (a highly impractical setup).

In order to deal with such shortcomings, we propose a decoupled viewpoint solution using SLAM technology for humanoid teleoperation. Our idea is novel and simple to

Yang Chen* and Leyuan Sun* are co-first authors and contributed equally to this work. Corresponding author: Yang Chen.

Y. Chen is with the School of Integrative and Global Majors (SIGMA), University of Tsukuba, Japan. chenyang@ai.iit.tsukuba.ac.jp

L. Sun, R.P. Singh and F. Kanehiro are with Department of Intelligent and Mechanical Interaction Systems, Graduate School of Science and Technology, University of Tsukuba, Japan.

G. Caron is with Universite de Picardie Jules Verne, MIS lab, Amiens, France.

All authors are with CNRS-AIST JRL (Joint Robotics Laboratory), IRL, National Institute of Advanced Industrial Science and Technology (AIST). son.leyuansun, mehdi.benallegue, rafael.cisneros, rohan-singh, k.kaneko, guillaume.caron, f-kanehiro @aist.go.jp; arnaud.tanguy, kheddar@lirmm.fr

K. Suzuki is with the Faculty of Engineering and Center for Cybernetics Research, University of Tsukuba, Japan. kenji@ieee.org

understand: we are using a point cloud as real-time visualization feedback, and a pre-constructed mesh that complement and enhance real-time visual data in blank areas caused by the previously mentioned shortcomings. The latter (tracking latency, robot's joint range limits and camera's smaller FOV) are handled all at once. This is illustrated in Fig. 1.

The main contributions of our paper are listed as follows:

1) We devised a decoupled viewpoint solution for a humanoid platform which allows the operators to sustain visual feedback changes with low-latency when they freely control their hidden view duplicate in an augmented reality environment.
2) Use a reconstructed mesh built by SLAM to complement the blank area caused by the mismatch between FOVs, or the mismatch between range of neck joints or the lag between operator's and robot's head motion.
3) We propose an online calibration solution which could align the virtual HMD and the virtual robot camera frame for the best visualization quality of the point cloud.

II. Related works

In this section, we introduce the SLAM technology used in humanoid robots and the different televisualizations in VR teleoperation with a focus on robot application.

SLAM in humanoids: Humanoid robots already use visual SLAM technology for navigation in various environments thanks to their embedded cameras, e.g. [3]. Recently, [4] used the map built by SLAM not only for locomotion but also for searching an object and estimating its pose by using point cloud registration. [5] integrated dense SLAM to a QP control framework for localization and also balancing a humanoid robot, which allows using visual odometry to make a reaching plan adjustable online. Besides, SLAM is a fundamental technology when humanoids are used in large-scale manufacturing settings, e.g., [6]. It is used to provide precise localization to a humanoid within its environment, build a semantic-reconstructed map, define the walking targets and so on. These few examples only show that SLAM is already adopted in several humanoid applications and is part of the humanoid basic planning and control architecture. This work highlights another application of the existing embedded SLAM in the humanoid embodied telepresence context.

With respect to televisualization works, we introduce a solution using first-person-view for televisualization, which is a common choice to be applied on humanoid platform, as it is more immersive than the third-person-view.

Stereo RGB fixed: Displaying real-time stereo RGB image from a stereo camera by means of a VR HMD [7], [8] is the most common practice in VR teleoperation. Compared with non-VR type, such as video streaming on an external monitor, the HMD allows better immersive capabilities and is more suitable to estimate the depth of objects present in the remote world. However, the latency between the motion of operator's and robot's head induced by a fast motion of the former, compromises the coherency of the visual feedback w.r.t sensory expectations and may cause dizziness. Moreover, this problem compromises intuitive interaction, as explained in Section I.

Decoupled viewpoint: Instead of directly mapping the stereo RGB image to the operator both eyes, decoupled viewpoint means that the operator could realize the an independent motion from the real-time robot's head motion in VR visualization. For example, when the operator moves the head quickly, the direct mapping solution does not update visual feedback instantly, yet a naive *decoupled viewpoint* will generate blank (empty) spaces in the image first and the RGB or point cloud will move towards that space gradually with the speed of the robot's head movement. Theoretically, this solution could reduce the dizziness compared with the direct mapping solution, since the operator could realize that the head motion changes the visual feedback, even if there is only a blank area being displayed. Yet, the latency in visual feedback display in the HMD still exists. The most recent work we could find which applied a decoupled viewpoint control is reported in [9], [10], and is developed by Team NimbRo in the frame of ANA Avatar XPrize. Thanks to the wide-angle camera that they use and the 6D DOF of their robot's head, their solution could allow the operator to freely control their perception in the VR device and the blank space caused by a *decoupled viewpoint* could be complemented by a spherical rendering method that they proposed. However, the limitation of spherical rendering is that the entire scene is considered with a constant depth, or otherwise, a distortion of image happens (as they reported).

Decoupled viewpoint with reconstructed mesh or CAD model: The closest solution we found related to this concept is from Team I-BOTICS[1] for ANA Avatar XPrize using RGB with a CAD model as televisualization feedback. The RGB image provides real-time visual feedback, while the CAD model of the scene complements the latency that causes the aforementioned blank area. Different from this solution, we rather use a point cloud to fuse with the reconstructed mesh built by using SLAM technology. The advantages of our solution are that i) we do not need the knowledge of the environment since the mesh can be reconstructed online when the robot explores the working area; and ii) the visualization of point cloud fused (aligned) with the mesh has better immersion properties compared to RGB with CAD and mesh; see Section III-C.

For other mobile platforms, a third-person-view is used. For example, in [11] the operator can adjust the pose of the viewpoint manually; they also provide a constructed mesh for visualization. However, there is no real-time information to complement the mesh, and the mesh is updated with a low frequency, making it difficult to handle dynamic scene. An intuitive interface for bimanual robot teleoperation system was developed in [12]. RGB merged with 3D point cloud is used for televisualization. The system could track the operator's head motion to make him freely look around using the VR view. However, the limitation of this solution is the need for additional operator support to select these views'

[1] https://www.youtube.com/watch?v=mL9SyGfuqI4

positions.

For teleoperation of industrial robotic arms, the third-person-view is the best choice, as a clear and complete observation is more important than immersion. [13] proposed three viewpoints for the operator to select during teleoperation. In the HMD, the operator could select freely to see the model of robot arm from side, front, and top view. Another approach called Picture-in-Picture (PiP) is also common, where video streams from multiple views are displayed concurrently and placed next to each other [14], [15]. Both solutions increase the cognitive burden for the operator. [16] provided a multi-visual source merging solution where the local point cloud is projected to a global stereo image nicely, providing additional information for grasping tasks. However, in contrast to our solution, it requires an additional visual sensor. In [17], the authors use a VR environment to compensate the difference of FOV between HMD and camera. It has better immersion than not using anything but the artificial environment cannot represent the true surroundings like our reconstructed mesh. So far, the early work of a total VR-based decoupling is reported in [18], but this solution cuts fully the operator from the real perception even if features such as the manipulated objects are updated from the real world.

III. Methodology

A. Robot Platform and VR Equipment

We use the HRP-4CR robot [19], an adult-size humanoid with realistic appearance, which is an upgraded and "refreshed" version of HRP-4C in [20]. A ZED Mini camera (from Stereolabs Inc.) is mounted on the forehead of the robot. On the operator side, an HTC Vive Pro Eye HMD (from HTC Corporation) is worn by the operator to receive the visual information and transfer the control command to the robot's head at the same time. The joystick which is mounted on a Valve Index controller (from Valve Corporation) is used for controlling the locomotion of the robot.

Our method is general enough to apply to any other similar set-up with humanoid, camera and HMD providing few coding adjustments.

B. System architecture

The robot control unit is an Intel NUC computer, a Jetson Nano (from Nvidia Corporation) is used for interfacing with the ZED Mini camera and streaming the visual information over a wired-LAN. The operator station is driven by a PC with specification (Intel i9-9900K 3.6 GHz, Nvidia RTX 2080Ti), mainly for running Unity3D (the platform we used for building a virtual space). All computers are connected by Ethernet. The system architecture is shown in Fig. 2. The communication protocol between NUC and Windows PC is ROS-Sharp[2].

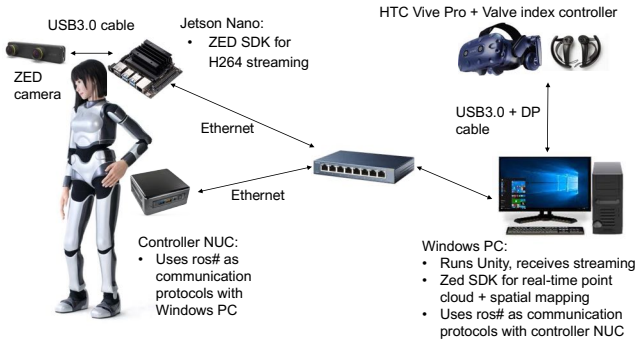

Fig. 2. System architecture.

C. Conceptual evaluation of visual expressions

Compared to the method that fuses 3D mesh and 2D RGB image, fusing 3D point cloud and 3D mesh would result (in theory) in a better merging visual effect. As a result, one expects a better immersion for the operator. The difficulty of fusing 2D RGB image and 3D mesh comes from the distortion of a 2D image when observed from different viewpoints. A concept of visualization is illustrated in Fig. 3, where there is a decoupled movement between the virtual HMD camera and the virtual robot camera in a VR environment (virtual space). Consider that the robot camera is capturing a 3D object such as a cube, then a RGB image or point cloud can be generated in the virtual sapce, due to the 2D property of the RGB image, the distortion can happen when it is observed from the virtual HMD. The distortion can be much less if 3D point cloud are used instead and if we assume the shape of point cloud is similar to the object itself.

The fusion quality between 3D point cloud and 3D mesh relies on the quality of the constructed mesh, the point cloud and the accuracy of odometry. Here we use a state-of-art 3D construction method called "Spatial Mapping" [3] and ZED visual-inertial odometry. With the odometry as a memory of the point cloud localization, the alignment between the constructed mesh and point cloud can be accomplished automatically. An expected fusion-effect example is shown in Fig. 1; compare the real scene (top side) and the virtual scene (left-down side), there we see that the mesh is well-fused with the real-time point cloud.

A fusion between the constructed mesh and the real-time point-cloud with an additional decoupled viewpoint control method will be presented in Section III-D. Using that method, the operator is able to intuitively control the perspective to observe the environment as s/he wants, increasing the immersion. However, in our tele-operation scenario, being aware of the difference between self and robot is important, since there exists a significant difference between a human and the avatar robot. Considering that the operator moves the head quickly, the perspective also changes fast. If the operator is not aware that the robot head motion is delayed, then the operation may cause an accident, e.g. a robot

[2]https://github.com/siemens/ros-sharp

[3]https://www.stereolabs.com/docs/spatial-mapping/

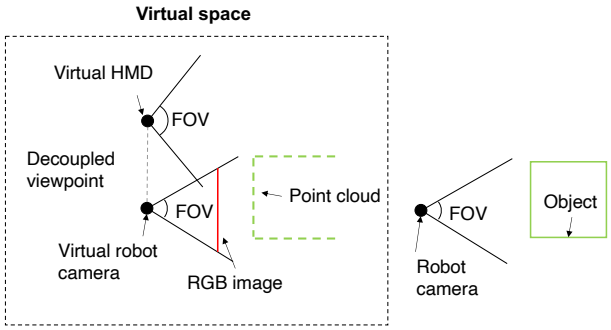

Fig. 3. Concept analysis. The virtual HMD should be able to see what the real robot camera can see, in the scenario denoted above, the distortion could happen.

collision with the environment. Our proposed solution is to make the color of the constructed mesh a bit different from the real color of the environment. Therefore, the operator can distinguish the mesh and real-time point-cloud to achieve a balance between awareness and immersion. This is also helpful when the environment is dynamic, as the operator would put more trust on the real-time information instead of the mesh.

D. Construction of virtual space & teleoperation mapping

The creation of a virtual space is necessary for the VR televisualization system which we introduced in Section II. From the solution of *Stereo RGB fixed* to *decoupled viewpoint with reconstructed mesh or CAD model*, the complexity of the virtual space increases. Here we explain the key difference between those solutions; specifically, we introduce the construction method of our proposal, which lies on the category of *decoupled viewpoint with reconstructed mesh or CAD model*.

For the *stereo RGB fixed*, the virtual space is the simplest situation: map the image of each camera eye to each eye of the HMD. For the *decoupled viewpoint*, we need to represent the decoupled motion of a robot camera and HMD in the virtual space. For the *decoupled viewpoint with reconstructed mesh or CAD model*, because we have the mesh as a world reference in the virtual space, not only the relative relation between robot camera and HMD has to be represented properly, but also the relation between robot camera, HMD and the mesh.

To better demonstrate the mapping relationship in our tele-visualization system, we have separated the space into three parts: operator space, virtual space, and robot space, as shown in Fig. 4. All frames are described in TABLE I.

The measurements are shown in Eq. 1: The description of the HMD frame in the tracking space frame is $M_H^T = M_B^T \cdot M_H^B$, where M_B^T is a fixed value calibrated when we did the VR room setup procedure, M_H^B is a real-time measurement made by the base station (from Valve Corporation), and M_H^T represents the head movement of the operator. The ZED pose relative to the robot body frame is $M_Z^R = M_{Rh}^R \cdot M_Z^{Rh}$, which are measured from the forward kinematics and the extrinsic respectively. M_Z^W is measured by the ZED visual-inertial odometry in world frame (using ZED SDK). Then,

TABLE I
Frame definition

Virtual space	Operator and Robot space
Virtual HMD frame: ***H'***	HMD frame: ***H***
Virtual ZED frame: ***Z'***	ZED frame: ***Z***
Virtual tracking space frame: ***T'***	Tracking space frame: ***T***
Virtual robot body frame: ***R'***	Robot body frame: ***R***
World frame in virtual space: ***W'***	World frame: ***W***
Mesh frame: ***S***	Robot's head frame: ***Rh***
Base station frame: ***B***	

$$\mathbf{M} = [M_H^T, M_Z^R, M_Z^W] \tag{1}$$

Those ones are also mapped from the operator and robot space to the virtual space as shown below:

$$\begin{aligned} \mathbf{M}^{'} &\overset{\text{def}}{=} \mathbf{M}, \\ \mathbf{M}^{'} &= [M_{H'}^{T'}, M_{Z'}^{R'}, M_{Z'}^{W'}]. \end{aligned} \tag{2}$$

In order to realize the decoupled motion between the virtual ZED frame and the virtual HMD frame, we rely on Eq. 3, namely

$$\begin{aligned} M_{Z'}^{H'} &= M_{T'}^{H'} \cdot M_{R'}^{T'} \cdot M_{Z'}^{R'} \\ &= M_T^H \cdot M_{R'}^{T'} \cdot M_Z^R, \end{aligned} \tag{3}$$

where $M_{R'}^{T'}$ is a static transformation which can be calibrated in order to display point cloud in HMD properly,. The details of the calibration will be discussed in Section III-E. When there is no mesh, $M_{Z'}^{W'} = M_{R'}^{W'} \cdot M_{Z'}^{R'}$, $M_{T'}^{W'} = M_{R'}^{W'} \cdot M_{T'}^{R'}$, and $M_{H'}^{W'} = M_{R'}^{W'} \cdot M_{T'}^{R'} \cdot M_{H'}^{T'}$ can be determined by any given constant value of $M_{R'}^{W'}$. As a result, we realize a flow: the operator moves the head $\rightarrow$ the robot's head follows this movement with a motion delay $\rightarrow$ the point cloud in virtual space moves together with the ZED frame with a network delay $\rightarrow$ the visual feedback described in the HMD frame updates accordingly.

However, when there is a mesh, to align point cloud and mesh properly, instead of being a constant value, $M_{R'}^{W'}$ should be the odometry of robot's body frame w.r.t. the world frame. Here, we rely on the ZED visual-inertial odometry, and use Eq. 4 to realize

$$M_{R'}^{W'} \overset{\text{def}}{=} M_R^W = M_Z^W \cdot M_Z^{R^{-1}}. \tag{4}$$

The precision of ZED visual-inertial odometry is studied in [21]. M_R^W can also be obtained by other localization methods such as motion capture or the kinematics-inertial odometry of the robot itself, but those methods either require additional sensor or they are not precise enough for our requirements. In the case of having a mesh, we have another flow for locomotion: the operator controls the robot's locomotion by joystick $\rightarrow$ the robot's body moves with a motion delay $\rightarrow$

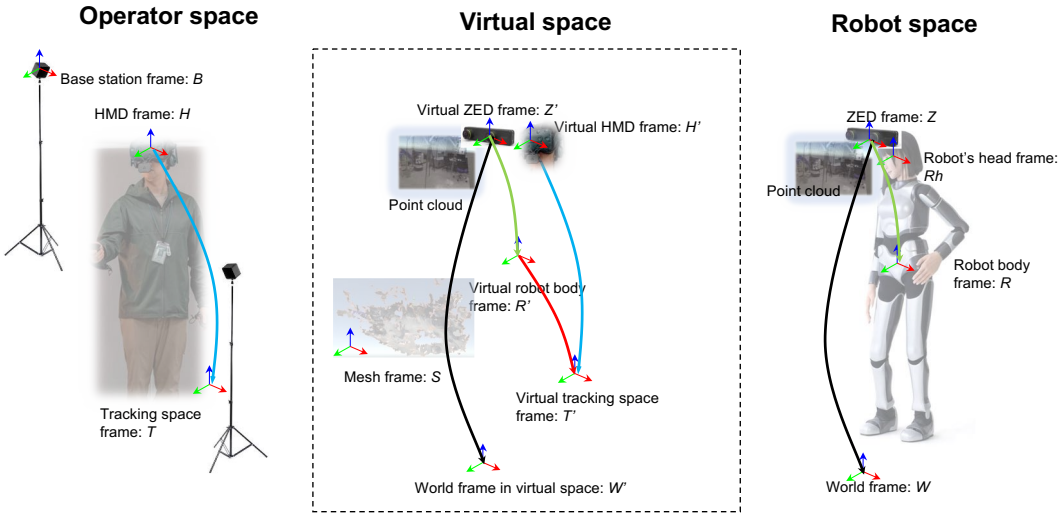

Fig. 4. Teleportation mapping: the motion of the operator's head, robot's head and ZED camera are mapped to the virtual space as depicted by the blue, green and black lines.

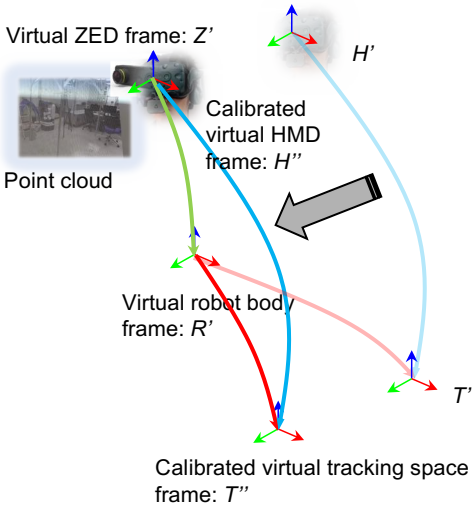

Fig. 5. Calibration process for the best viewpoint of the point cloud visualization: move the virtual tracking space frame from T' to T'' in order to align H'' with Z'. H'' and T'' are the calibrated virtual HMD frame and calibrated virtual tracking space frame during the calibration process. After the calibration, H'' coincides with Z'.

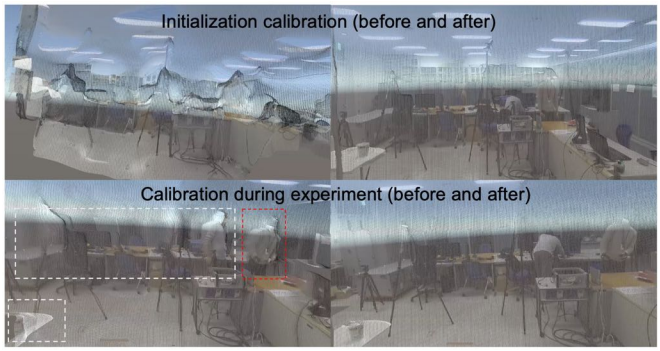

Fig. 6. Top: initialization calibration, bottom: calibration during experiment, distortion shown in white rectangle caused by the operator shifting position as shown in the red rectangle. Left represents before, right represents after calibration.

the point cloud in virtual space moves together with the ZED frame with a network delay $\rightarrow$ the visual feedback described in the HMD frame updates accordingly.

E. Online calibration

We have realized the decoupled motion control for both virtual ZED frame and virtual HMD frame by using Eq. 3; however, the height of users can be different, or the user may also move the body slightly during operation. Both factors cause a change of $M_{H'}^{T'}$, resulting in the virtual HMD frame misalignment with respect to the virtual ZED frame. The visualization of point cloud from different viewpoints should have a similar effect as we analyzed in Section III-C. However, as the point cloud is usually not generated perfectly, its visualization from an offset viewpoint might cause a distortion as shown in Fig. 6. Furthermore, the misalignment can cause the operator to misjudge the relative pose between objects and robot. In order to visualize the point cloud correctly inside the FOV of the HMD, we propose the online calibration method shown in Fig. III-E. The calibration problem is defined as bellow:

$$\begin{aligned} & M_{H''}^{W'} = M_{Z'}^{W'}, \\ \text{s.t.} \quad & M_{T''}^{H''} = M_{T'}^{H'}, \end{aligned} \tag{5}$$

where H'' and T'' represent the calibrated virtual HMD frame and calibrated virtual tracking space frame. The relative pose between the virtual HMD frame and the virtual tracking space frame is always decided by the localization based on the base station's measurement; so instead of changing the pose of the virtual HMD frame directly, we move the virtual tracking space frame from T' to T'' to align the calibrated virtual HMD frame with the virtual ZED frame as decomposed by Eq. 6:

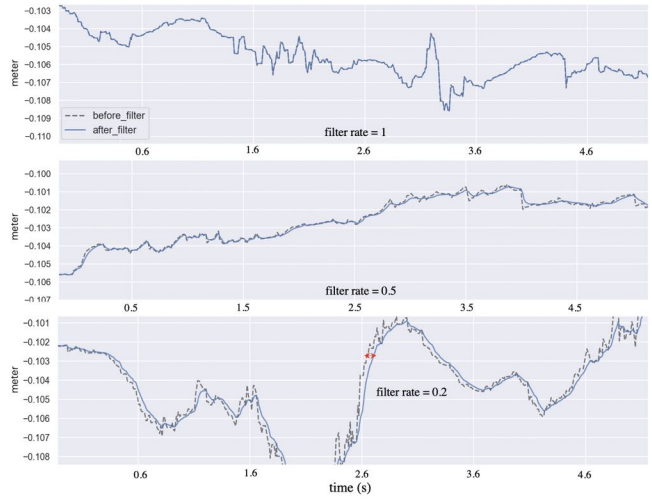

Fig. 7. The latency caused by different rates of the low-pass filter.

$$M_{H''}^{W'} = M_{R'}^{W'} \cdot M_{T''}^{R'} \cdot M_{H''}^{T''} = M_Z^W \cdot M_Z^{R^{-1}} \cdot M_{T''}^{R'} \cdot M_H^T. \quad (6)$$

We can obtain transformation from the T'' to R' by using Eq.7, which is the one we need to update the alignment:

$$M_{T''}^{R'} = M_{Z'}^{R'} \cdot M_{T''}^{Z'} = M_Z^R \cdot M_{T''}^{H''} = M_Z^R \cdot M_T^H. \quad (7)$$

One example of the effect of calibration is shown in Fig. 6. There are also more things to note: (1) The calibration can be activated by the operator voluntarily pressing a trigger on the controller during tele-operation. (2) The calibration usually only needs to be done when the operator has changed or when the operator's position changes too much during the tele-operation process, as most of the time the point cloud looks good enough even with a small misalignment. (3) The calibration can be activated at any moment and can finish instantly during the operation. (4) It's better to calibrate with a looking forward posture and in a static status, with consideration for the best effect on the decoupled viewpoint.

F. Low Pass Filter

Like other related works [9] [10] and [11], we also need to apply a low-pass filter in order to increase the smoothness of motion and immersion of VR experience. Before its implementation, we first analyze the source of noise. The noise resulting from observing the point cloud from the HMD could be analyzed by using Eq. 3, where all three values are stable measurements. M_T^H and $M_{R'}^{T'}$ are local measurements on the Unity side, while M_C^Z is measured on the controller PC and sent by Rosbridge communication protocol through Ethernet to the Unity side. The noise arises mostly from the last one due to the possible instability of Rosbridge communication and the network.

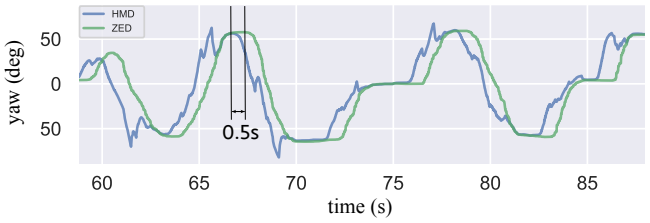

Fig. 8. Common latency between the motion of operator's and robot's head: approx. 0.5 s.

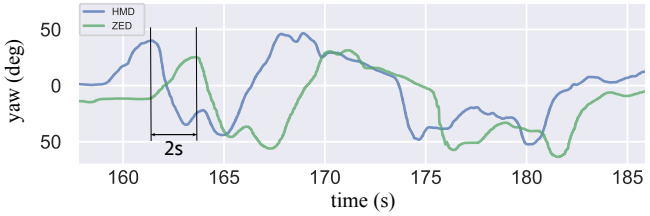

Fig. 9. Latency when the network was unstable: approx. 2 s.

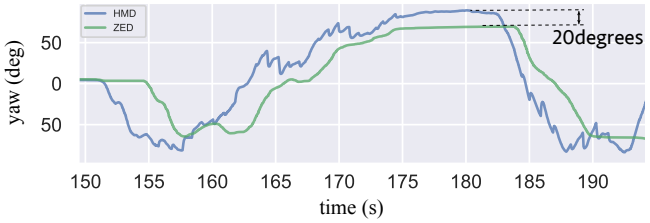

Fig. 10. Range of motion measurement

The noise resulting from observing the mesh from the HMD can be analyzed by Eq. 8. Compare it with Eq. 3. Here, $M_S^{W'}$ is a static measurement on the Unity side, M_W^Z is the ZED odometry measurement sent through network but not by Rosbridge. The noise resulting from observing the mesh could be larger than observing the point cloud due to the usage of the ZED odometry.

$$M_S^{H'} = M_{W'}^{H'} \cdot M_S^{W'} = M_T^H \cdot M_{R'}^{T'} \cdot M_Z^R \cdot M_W^Z \cdot M_S^{W'} \quad (8)$$

In order to reduce the dizziness of the operator caused by the noise, we apply a low-pass filter to the virtual tracking space frame.

In Fig. 7, we plot the comparison between the measurement before and after applying the filter using different filter rates. We observe that the high frequency component is removed and when we decrease the rate, the curve becomes smoother, while the lag increases. In our experiment, we use a filter rate of 0.2, such that the latency caused by the low-pass filter is approx. 60 ms. In the future, we will use a filter without delay, such as the complementary filter [20], as the performance is more suitable for VR.

IV. Experiment

A. Latency Measurement

We measured the main latency in the system between the motion of operator's and robot's head. We could observe

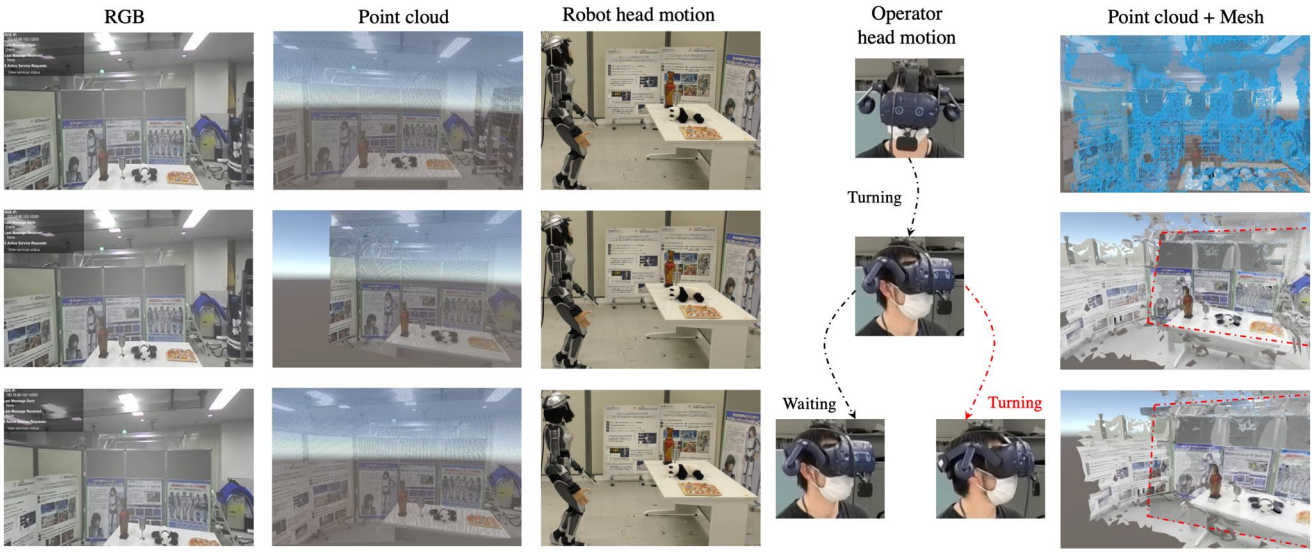

Fig. 11. Three solutions were compared during the experiment. In the first, second and last columns, we show the image displayed in the HMD for the three different solutions. For the point cloud with mesh (last column, first row), the blue grid shows the user scanned area while the mesh is under construction. In the second and third rows, we marked the real-time point cloud by a red dot line. In the fourth column, from top to down, the displayed images correspond to different operator's motions. In cases of RGB and point cloud, the operator turned to the left from static forward status and then waited a bit for the real-time visual feedback. While in case of the point cloud with mesh, the operator kept turning. The third column shows the robot's head motion, and due to the lag, the robot' head motion does not change in the second row, although the operator had turned his head.

these two motions in Unity3D which is the software where our virtual space is built. We ask the operator to turn the head from left to right (approx. -75 deg to 75 deg). In Fig. 8, usually when network is stable, we could see the latency is about 0.5 s. However, the latency could increase to 2 s as depicted in Fig. 9 when the network became unstable.

B. Range of Motion Measurement

One advantage of our system is that it is able to compensate the blank area caused by the mismatch between the range of motions of human's and robot's heads. The motion of both ZED and HMD are shown in Fig. 10. As expected, after the operator approached the maximum angle of head motion, there were 20 deg of gap between the human's head and robot's head in a static situation. In case of the *stereo RGB fixed* solution, the operator would lose this information of the environment. In case of our solution, the operator could see the pre-built mesh. The qualitative result is shown in the next section.

C. Qualitative Evaluation

We compared our proposed solution with a common *stereo RGB fixed* solution which is adopted by most of teams in the XPRIZE competition such as [22] [23] as qualitative evaluation. Since our proposal mainly include two parts: one is the decoupled representation in a virtual space, another one is to overlap the point cloud with mesh, we evaluate this two parts separately. Seven participants (males between their 20s and 40s) were asked to finish one small task three times. The task was to tele-operate the robot to walk straight forward or backward freely and, at the same time, they could turn the head freely to observe the environment as they wanted. For each trial, different visual feedback solutions were provided: stereo fixed viewpoint with RGB image only, decoupled viewpoint with point cloud only or decoupled viewpoint with point cloud and mesh, which correspond to the three solutions that we explained in Section II. The experiment scene is shown in Fig. 11. For the second and third solutions, the participants were able to calibrate the viewpoint at any time during the operation. For the third solution, the participants were asked to build the mesh right after we started the system, and they could scan the area as much as they wanted. After building the mesh, the point cloud always overlapped the constructed mesh. In order to reduce the burden to the network, we set the point cloud resolution as HD720; however, we also include the performance of point cloud HD1080 in a demo video [4].

The comparison snapshots of the three solutions are shown in Fig. 11. From the solution of point cloud and point cloud with mesh, we could observe that when the operator turned his head, the real time point cloud could track the center of the HMD display with a delay. In the solution of point cloud with mesh, we could see the fusion between the point cloud and the constructed mesh, as well as that the mismatch between the ZED's and HMD's FOV is complemented by the pre-built mesh. One interesting thing we found is that since the reconstructed mesh could complement the blank area, the operators usually did not wait for the real-time point cloud to come back to the center of the HMD's FOV, but they would keep intuitively moving their perspective, which significantly increased the interaction efficiency with robot. This fact is represented in Fig. 11: after turning the head a small angle, the operator kept turning his head to a larger degree until he saw the blank area which was not covered

[4] https://www.youtube.com/watch?v=Jdiaosp_qH8

by mesh (also beyond the motion range of the robot's head) as the last figure shows in the last column.

V. Conclusion and future work

In this work, we proposed a balanced-immersive televisualization solution with decoupled viewpoint control and complementary visual feedback for humanoid robot teleoperation. The proposed solution mainly contains two core ideas. The first idea is to complement the real-time point cloud with a constructed mesh, aligning them by applying visual-inertial odometry while distinguishing them by changing the color of the mesh, resulting in a balanced fusion effect. The second idea is to realize the decoupled movement between robot's and operator's viewpoints in a virtual space to reduce the visual latency. We achieved a balanced immersion by combining both ideas. We evaluated both ideas by comparing against a standard stereo RGB fixed view solution and verified the effectiveness of our proposal regarding the provision of instant visual feedback, bigger FOV and bigger range of view (due to a bigger range of human head motion). We also identified several limitations that could be improved in the future. One limitation comes from the quality of the reconstructed mesh and the odometry precision leading to a misalignment between point cloud and mesh, which can be solved by a continuously improved higher precision SLAM method, available nowadays. Also, the real-time part should always be given a higher visual priority than the constructed mesh. This could be achieved by using other rendering methods. Another future work will be evaluate the proposed system with a complete user study evaluated by questionnaire. Our proposal is most suitable for a humanoid platform, but potentially it could also be applied to other robotic platforms, especially those using human head motion to control a robot's head with a mounted camera.

References

[1] 2022 XPRIZE Foundation, "Ana avatar xprize," https://www.xprize.org/prizes/avatar, Last accessed on 2022-07-15.

[2] J. Zhao, R. S. Allison, M. Vinnikov, and S. Jennings, "The effects of visual and control latency on piloting a quadcopter using a head-mounted display," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2018, pp. 2972–2979.

[3] A. Hornung, K. M. Wurm, and M. Bennewitz, "Humanoid robot localization in complex indoor environments," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 1690–1695.

[4] M. Tsuru, A. Escande, A. Tanguy, K. Chappellet, and K. Harada, "Online object searching by a humanoid robot in an unknown environment," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2862–2869, 2021.

[5] A. Tanguy, P. Gergondet, A. I. Comport, and A. Kheddar, "Closed-loop RGB-D SLAM multi-contact control for humanoid robots," *2016 IEEE/SICE International Symposium on System Integration (SII)*, pp. 51–57, 2016.

[6] A. Kheddar, S. Caron, P. Gergondet, A. Comport, A. Tanguy, C. Ott, B. Henze, G. Mesesan, J. Englsberger, M. A. Roa, P.-B. Wieber, F. Chaumette, F. Spindler, G. Oriolo, L. Lanari, A. Escande, K. Chappellet, F. Kanehiro, and P. Rabaté, "Humanoid robots in aircraft manufacturing: The airbus use cases," *IEEE Robotics Automation Magazine*, vol. 26, no. 4, pp. 30–45, 2019.

[7] J. Nakanishi, S. Itadera, T. Aoyama, and Y. Hasegawa, "Towards the development of an intuitive teleoperation system for human support robot using a VR device," *Advanced Robotics*, vol. 34, no. 19, pp. 1239–1253, 2020.

[8] C. Wang, X. Chen, Z. Yu, Y. Dong, R. Zhang, and Q. Huang, "Intuitive and versatile full-body teleoperation of a humanoid robot," in *2021 IEEE International Conference on Advanced Robotics and Its Social Impacts (ARSO)*, 2021, pp. 176–181.

[9] M. Schwarz, C. Lenz, A. Rochow, M. Schreiber, and S. Behnke, "NimbRo avatar: Interactive immersive telepresence with force-feedback telemanipulation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 5312–5319.

[10] M. Schwarz and S. Behnke, "Low-latency immersive 6D televisualization with spherical rendering," in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, 2021, pp. 320–325.

[11] P. Stotko, S. Krumpen, M. Schwarz, C. Lenz, S. Behnke, R. Klein, and M. Weinmann, "A VR system for immersive teleoperation and live exploration with a mobile robot," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 3630–3637.

[12] T. Rodehutskors, M. Schwarz, and S. Behnke, "Intuitive bimanual telemanipulation under communication restrictions by immersive 3D visualization and motion tracking," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, 2015, pp. 276–283.

[13] A. Naceri, D. Mazzanti, J. Bimbo, D. Prattichizzo, D. G. Caldwell, L. S. Mattos, and N. Deshpande, "Towards a virtual reality interface for remote robotic teleoperation," in *2019 19th International Conference on Advanced Robotics (ICAR)*, 2019, pp. 284–289.

[14] Y.-T. Lin, Y.-C. Liao, S.-Y. Teng, Y.-J. Chung, L. Chan, and B.-Y. Chen, "Outside-in: Visualizing out-of-sight regions-of-interest in a 360 video using spatial picture-in-picture previews," in *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, 2017, pp. 255–265.

[15] D. Whitney, E. Rosen, E. Phillips, G. Konidaris, and S. Tellex, "Comparing robot grasping teleoperation across desktop and virtual reality with ros reality," in *Robotics Research*. Springer, 2020, pp. 335–350.

[16] D. Wei, B. Huang, and Q. Li, "Multi-view merging for robot teleoperation with virtual reality," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8537–8544, 2021.

[17] D. Sun, A. Kiselev, Q. Liao, T. Stoyanov, and A. Loutfi, "A new mixed-reality-based teleoperation system for telepresence and maneuverability enhancement," *IEEE Transactions on Human-Machine Systems*, vol. 50, no. 1, pp. 55–67, 2020.

[18] A. Kheddar, "Teleoperation based on the hidden robot concept," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 31, no. 1, pp. 1–13, 2001.

[19] R. Cisneros-Limó, M. Benallegue, K. Kaneko, H. Kaminaga, G. Caron, A. Tanguy, R. Singh, L. Sun, A. Dallard, C. Fournier, M. Tsuru, C. Yang, Y. Osawa, G. Lorthioir, F. Kanehiro, and A. Kheddar, "Team JANUS humanoid avatar: A cybernetic avatar to embody human telepresence," in *RSS 2022 Workshop on "Towards Robot Avatars: Perspectives on the ANA Avatar XPRIZE Competition"*, 2022.

[20] K. Kaneko, F. Kanehiro, M. Morisawa, K. Miura, S. Nakaoka, and S. Kajita, "Cybernetic human HRP-4C," in *2009 9th IEEE-RAS International Conference on Humanoid Robots*, 2009, pp. 7–14.

[21] A. Alapetite, Z. Wang, J. P. Hansen, M. Zajaczkowski, and M. Patalan, "Real-world comparison of visual odometry methods," 2020.

[22] B. Park, J. Jung, J. Sim, S. Kim, J. Ahn, D. Lim, D. Kim, M. Kim, S. Park, E. Sung, H. Lee, G. Park, J. Cha, J. Shin, and J. Park., "Team SNU's avatar system for teleoperation using humanoid robot: ANA Avatar XPRIZE competition," in *RSS 2022 Workshop on "Towards Robot Avatars: Perspectives on the ANA Avatar XPRIZE Competition"*, 2022.

[23] J. M. C. Marques, N. Patrick, Y. Zhu, N. Malhotra, and K. Hauser, "Commodity telepresence with the AvaTRINA Nursebot in the ANA Avatar XPRIZE semifinals," in *RSS 2022 Workshop on "Towards Robot Avatars: Perspectives on the ANA Avatar XPRIZE Competition"*, 2022.