Robot-Human Interaction through a Humanoid Telepresence System

Carlos Girard[1], Diego Calderon[1], Ali Lemus[1], Victor Ferman[2] and Julio Fajardo[1,2]

***Abstract*— Both motion capture and mapping systems have been evolving to enhance the virtual immersion experience in a human-in-the-loop model. In this work, a motion mapping system compound by a 3D printed humanoid robot built with low-cost materials, an IMU-based motion capture suit, binaural microphones and a virtual reality headset is presented. The movement of the robot is limited by its reduced amount of degrees of freedom and by its shoulder configuration, that differs from the human's own biomechanics. With the motion capture suit's information, a ROS-based architecture maps these orientations to obtain the generalized coordinates of the robot in order to imitate the operator's arms motion. Additionally, the headset is used to project a stereo vision of the robot's surroundings and to maps the operator's head motion, as well as microphones, placed on the robot's head, to provide a more immersive experience and the ability to capture 3D sound. This project intends to provide an interactive telepresence puppetry system to encourage the involvement of a targeted audience on engineering subjects. The system shows precise results with moderate time response and satisfying a trade-off between cost and functionality.**

I. INTRODUCTION

Motion control in the field of humanoid robots has been approached with many techniques that involve a vast variety of motion tracking and motion detection implementations. Since the beginning of robotics, the topic of developing human shaped robots has been a great challenge for engineering, because the problems that involve replicating human movements, as well as the right acquisition of motion measurements from a human. Therefore, a lot of algorithms and devices had to shape the path for stable and robust systems that can accomplish these goals with acceptable results. One of the first attempts to harness humanoid control was by developing the inverse kinematics models and motion capture (MoCap) systems using cameras [1]. Moreover, having a control system to determine the position of the end effector of a robot is essential for critical applications like surgery assistance where there are human lives at risk [2]. In other applications, such as teleoperated robotic arms, exist a greater interest on replicating the orientation of the robot links rather than track the cartesian coordinates of the end effector, all this, in the sake that the operator can work more conveniently and intuitively [3].

Furthermore, as telepresence continues expanding to new applications, humanoid robots have been recently added to more immersive forms of interacting with humans.

[1] Authors are with Turing Research Laboratory, FISICC, Universidad Galileo, Guatemala City, Guatemala, email: {prospektrgirard,diegocdl,alilemus}@galileo.edu

[2] Authors are with Department of Computer Engineering and Industrial Automation, FEEC, UNICAMP, Campinas, SP, Brazil, e-mail: {vferman,julioef}@dca.fee.unicamp.br

The use of virtual reality headsets can provide the user a wide view of the information received by the robot's eye-cameras [4]. However, with this principle, comes the problem of latency from the stereo images perceived by the human eyes, producing motion sickness if the frame rate is not sufficient or if the images are not well synchronized. Hence, the proper selection of image resolution and frame rate can mitigate those problems [5].

In previous implementations, humanoid robots were designed as attractions on thematic parks with the aim to resemble the human movements triggering the intention to make a new kind of human-like puppets that can be operated with body movements. Consequently, the methods to acquire human actions where always following the path of using cameras with certain track points around human body or using motion sensor suits [6,7]. Both approaches have been constantly improved to achieve better and more robust modes of operation. Furthermore, the entertainment industry remains mostly interested in teleoperation and MoCap systems, since the video games and thematic parks are significantly profitable and direct significant resources to generating more research opportunities [8,9]. Otherwise, the use of MoCap systems has grown in other fields such as physiotherapy, where a humanoid robot can provide mechanical assistance to patients [10,11]. The use of IMU based MoCap systems brings benefits like better flow of information and an enhanced representation of each joint (once well calibrated), due the movement is mapped by a mechanical action and not inferred by a visual model [12].

After achieving a stable motion mapping system, it was possible to think of applications for education and make it accessible for low-income universities to have affordable humanoid robots for their students [13]. On the other hand it is also reasonable to think new solutions for thematic parks or develop human-robot collaborative environments that promote robotic services, aided by virtual reality environments that delivers a more vivid experience.

The concept of using humanoid robots to involve people on the robotics field, has always awakened curiosity and desire of develop more interactive projects of engineering. As a matter of fact, virtual reality devices like the Vibe, HoloLens and Oculus Rift have brought great solutions to head motion tracking and stereo vision, this type of devices have become mainstream, not only for entertainment but also for research applications [14]. Taking the case of the Oculus Rift, it offers an easy solution to obtain the head orientation of a human operator and almost directly send it to a robot head, not only to imitate its motion, but also to provide a solution based on a telepresence system.

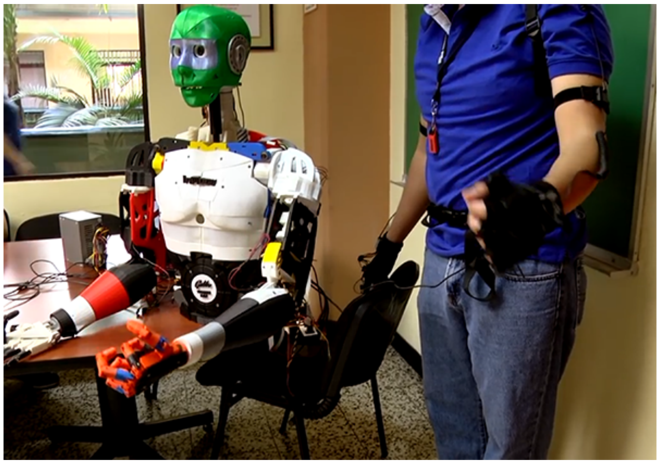

Fig. 1. The robot with an operator wearing the motion capture system.

This work presents Leonardo GreenMoov, a telepresence humanoid robot that is controlled through a MoCap suit, built with low-cost materials and based on the InMoov project [15]. Furthermore, the robot mapping problem for this particular case is solved by the use of forward and inverse kinematics techniques to deal with the differences generated by the joint configuration between the robot's and operator's arms. Moreover, the head of the robot has two high definition cameras whose images are projected in the Oculus Rift DK2 headset, alongside with binaural microphones, to provide the teleoperator the sensation of immersion in the environment where the robot is. All this, with the aim to bring a more personalized experience interacting with a targeted audience (in both ways, actor and audience) through a telepresence puppetry system [16], guiding and entertaining people in places like universities, theaters, thematic parks, malls and museums, in counterpart with the role of automated receptionist such as works described in [17,18].

The rest of this work is organized as follows, the Section II describes the functions and components that are part of the telepresence system. In Section III, forward and inverse kinematics are explained to give a better understanding of how the problem of robot mapping was approached. Finally, the Sections IV and V list the experimental results and conclusions obtained through simulations and with the use of the system in real scenarios.

II. System Architecture

A modular system is proposed in order to control the robot remotely, as well as enabling or disabling specific modules depending on the user needs (teleoperated or automatic mode). Its modularity also permits connecting multiple operating systems (OS). First, the system is composed by the Axis Neuron Pro (Motion Capture System Software), which is the only way to extract the information from the suit and that it is only available for Windows and macOS, the first option was chosen to keep the cost as low as possible. The humanoid robot uses a Pine A64 2GB single-board computer (SBC) as master controller, that runs nodes of Robot Operating System (ROS) to implement the robot mapping and the Leonardo's main controller, this SBC was selected because it is affordable and has low energy consumption. For an easy development, the system also includes an RViz simulation, which allows to run tests without the need to have the robot connected. Besides, three microcontrollers (MCUs) are connected to control the actuators of the physical robot through three point to point UART ports by the use of *rosserial* package. The Perception Neuron Suit is connected to a Windows PC using a USB port or WiFi over TCP/UDP protocols, in this way, the Pereception Neuron Pro software acquiere the packages and send them to the master controller using a TCP broadcast node. Then, all the modules interact through ROS topics with the aim to update the servomotors' positions using Pulse Width Modulation (PWM) generated by the MCUs.

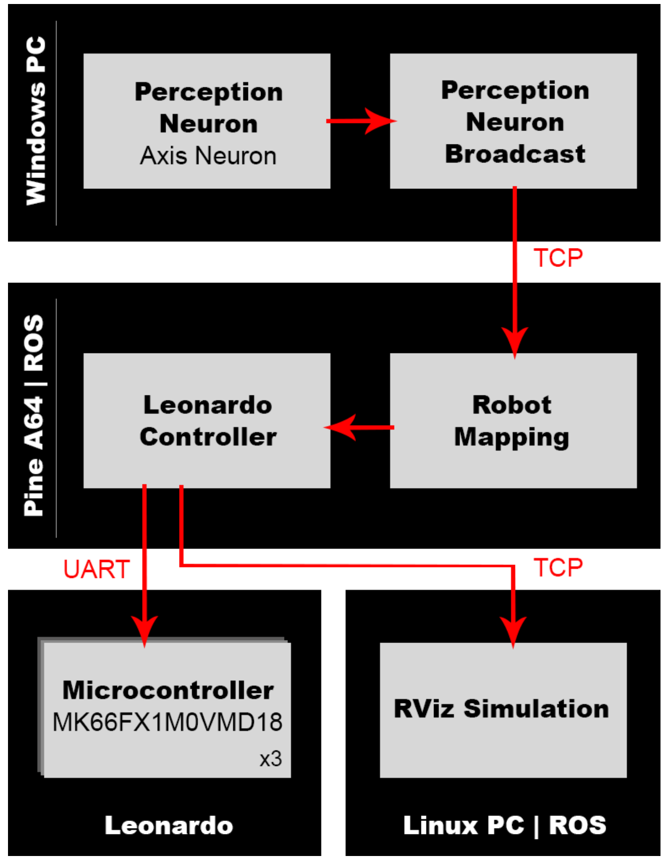

Fig. 2. Block diagram of system architecture implemented.

A. Perception Neuron

The Perception Neuron is a full-body MoCap developed by Noitom Ltd and that is composed of "neurons", which consist of Inertial Measurement Units (IMU) that are located in specific poses of the suit. These neurons are connected to a HUB that collects, process and packages information about the motion measured from the operator, transferring it at a frequency about 60 Hz, in the full body configuration. The firmware of the suit allows the development of third party software using the sensors' data through a C/C++ API.

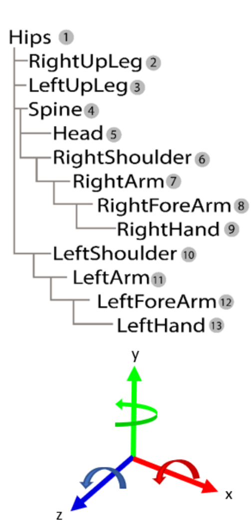

Fig. 3. The BVH format diagram for the Perception Neuron data.

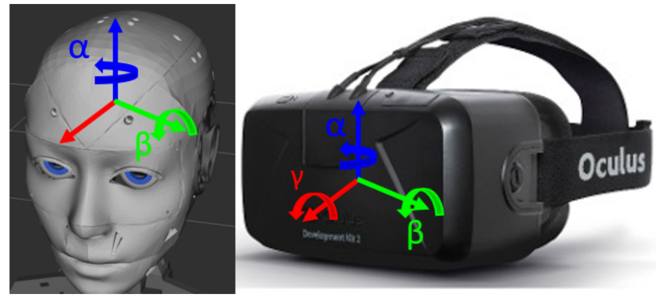

Fig. 4. Representation of coordinate frames for the Oculus Rift headset and for the robot's head. x axis in red, y in green and z in blue.

The Axis Neuron Pro acts as a TCP server, while the Perception Neuron Broadcast node imports the C/C++ API to establish a connection to this server. The API provides a callback function that is called every time the server publishes a new data frame with information about motion represented with the Biovision Hierarchical format (BVH). This is a hierarchical format for representing MoCap data that is commonly used in the industry of animation and production of movies. It is composed of bones, where the body is described in a tree structure using Euler angles of each bone [19]. In this case, the "Hips" represent the ROOT of the tree, as illustrated in Fig. 3. The Perception Neuron Broadcast uses *rosserial_windows* (over TCP) to connect to the ROS core that is running into the Pine A64. Every time the callback function in the Perception Neuron Broadcast is called, it takes the data frame and sends it as ROS topics for its use in the robot mapping node.

B. Oculus Rift

The Oculus Rift is a popular Virtual Reality (VR) headset used in order to view what the robot is seeing through its eye-cameras bringing an immersing experience to the users [20]. This is achieved using two HD cameras placed into the eyes of the robot capturing video that is processed and projected in the Oculus display. Furthermore the headset is used to imitate the operator's head with the robot's head, complementing the generalized coordinates provided by the MoCap system, as shown in Fig. 4. This is obtained using the Oculus SDK using a Windows PC. Then, the information about its Euler angles of the operator's head is sent to the Leonardo controller in a similar way as the MoCap data.

C. Hardware Architecture

Leonardo GreenMoov is a 3D-printed humanoid robot based on the open source InMoov project [21], it has 51 degrees of freedom (DOF), 27 degrees of actuation (DOA), five for each hand (one per finger), five for each arm, two for the torso, two for the hips, two for the neck and one for the jaw. In this manner, the DOAs that are used for this project are the only ones involved to actuate the arms and the head, a total of 23 servo motors (5 in each arm, 3 in the head and 2 in the torso) that are controlled using three MK66FX1M0VMD18 MCU from NXP. Besides, it has two cameras in the eyes, a speaker and a set of binaural microphones placed in the robot's ears.

D. Motion Mapping Controller

The Robot Mapping is implemented in a ROS node that subscribes to the topics which has the Euler angles that are sent from the MoCap System and the VR headset. This node processes each message as is described in the Section III. Then, sends commands to the Leonardo Controller to execute the movements that the operator is performing. This controller acts as a central module of communication for the physical and the RViz simulation (Fig. 6), it receives the motion commands and validates them against the constraints of the physical robot to ensure that the movement will not break the robot for trying to move to an unreachable pose.

III. Motion Mapping

Because the motion capture suit is based on nodes composed by IMUs, the orientation of each bone/link are represented using Euler angles. To define the movements of the arms it was necessary to properly allocate the rotation axes in the robot model to have a consistent representation of each DOF. Since the robot needs to perform tasks that requires tracking on the links' orientation and without considering the end effector's position, it was not possible to relay on methods that are only sensitive to variations in that position, like damped least squares (DLS), rather than orientation [22,23]. In order to properly map the orientation of each joint from the operator's arms, it is imperative to resolve the inverse kinematics problem for the robot's arms joints to have a more complying performance. Since there are many options to solve this problem, the most effective one was to manually obtain the equations that describe the rotation of each joint of the robot's arms. To describe the orientation of the shoulder and elbow of the robot, it was necessary to generate the homogeneous transformation that indicates the orientation of the end effector of the arm through a rotation matrix, as it is shown in the Equation (1).

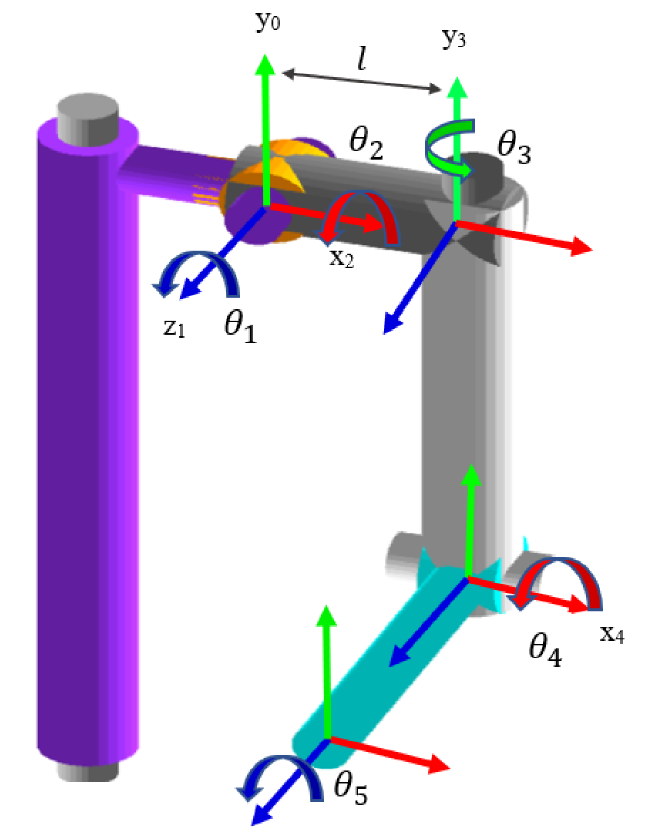

Fig. 5. Leonardo's arms geometry model with its coordinate frames.

In order to obtain the equations that map the generalized coordinates $\mathbf{q}$ of the robot's arm configuration (shoulder and elbow, as shown on Fig. 5), from the proper sequence of Euler angles obtained from the MoCap suit. We use the traditional notation from the defined rotation matrix $\mathbf{R_{03}}$ described in the equation (1), this matrix transforms the orientation of the elbow from the coordinate frames 0 (shoulder) to 3 (elbow), and that is composed by three column vectors named: normal, slide and approach where $\mathbf{R}_{03} = [\mathbf{n}, \mathbf{s}, \mathbf{a}]$, with $\mathbf{R}_{03} \in \mathbb{R}^{3x3}$, $\mathbf{n} \in \mathbb{R}^3$, $\mathbf{s} \in \mathbb{R}^3$ and $\mathbf{a} \in \mathbb{R}^3$. Namely, each one of these vectors have an x, y and z component. This vectors compose a 3×3 matrix that describes each one of the rotations that the robot executes as shown as follows:

$$\mathbf{R}_{03} = \begin{bmatrix} C_1C_3 - S_1S_2S_3 & -S_1C_2 & S_1S_2C_3 \\ S_1C_3 + C_1S_2S_3 & C_1C_2 & S_1S_3 - C_1S_2C_3 \\ -C_2S_3 & S_2 & C_2C_3 \end{bmatrix} \quad (1)$$

The presence of the spherical joint that humans have in the shoulder makes it very difficult to map movements directly to the actuators in a humanoid robot. At first, is necessary to perform the forward kinematics (FK) from the robot's arm configuration (shoulder to elbow) to obtained the rotation matrix previously mentioned. Therefore, the orientation of each frame was delimited with the Euler angles sequence using the Tait-Bryan convention ZXY, and it is also relevant to mention that there was no direct mapping configuration to match the proper Euler angles of the MoCap suit.

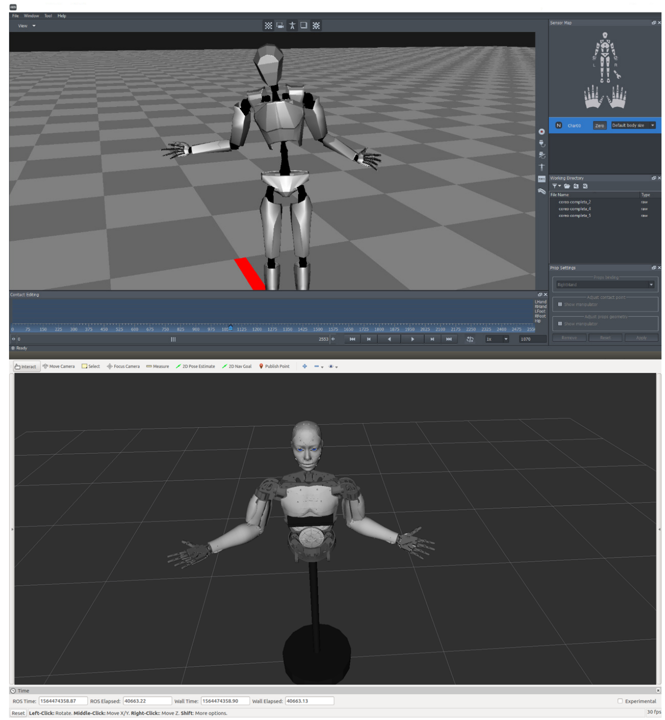

Fig. 6. Axis Neuron visualization and the RViz Simulation.

Also, the forward kinematics (shoulder to elbow) from the MoCap suit is necessary in order to execute the motion mapping properly by obtaining the orientation of the elbow, since the human body does not have a displacement l on the shoulder like the robot does. Therefore, this grants having an effective way to map the controller actions, since this is also an open loop system that does not have a feedback from the links position. This is the fundamental reason for which the system does not qualify as a control for human operations, instead it is a robot mapping solution that centers in imitating the arms' orientation within the robot's workspace.

The generalized coordinates for the robot's arm configuration are presented in the Equation (2). After knowing the orientation of the elbow, it was possible to directly map the rotations of elbow and wrist of the MoCap system using the following Euler angle sequence ZYX $= [\alpha, \beta, \gamma]$, where $\theta_4 = \gamma$ and $\theta_5 = \alpha$. This also relieves the system from singularities, hence there is no link configuration to such case. Fig. 5 illustrates which part of the arm is controlled with each one of the angles derived from the rotation matrices and also is the reference to elaborate the forward kinematics from the robot. As can be seen in Fig. 3, the same axis frame is used to generate the forward kinematics of the MoCap and having a better fit with the further calculations for the robot's arm orientation.

$$\mathbf{q} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \end{bmatrix} = \begin{bmatrix} \operatorname{atan2}(-\mathbf{s}_x, \mathbf{s}_y) \\ \sin^{-1}(\mathbf{s}_z) \\ \operatorname{atan2}(-\mathbf{n}_z, \mathbf{a}_z) \\ \gamma \\ \alpha \end{bmatrix} \quad (2)$$

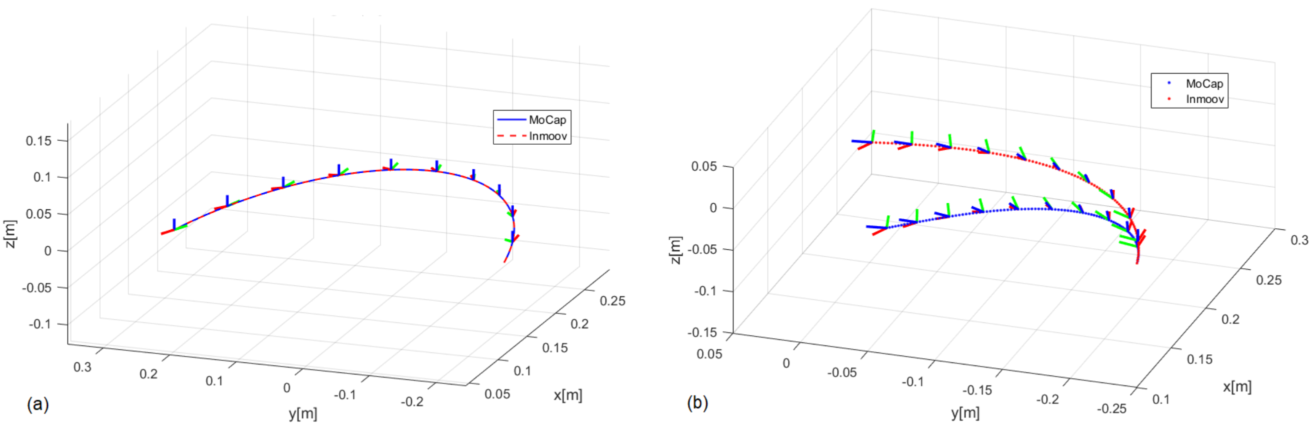

Fig. 7. Simulations results showing two different trajectories of the Leonardo's left elbow.

The fact of keeping the axes in the same orientation makes a simpler implementation than using more general approaches, like the Denavit-Hartenberg algorithm, that grants a particular solution that can be generated by aligning the reference axes in any other orientation, but this may produce further complications at the moment of assigning a positive or a negative rotation to the robot actuators. On the other hand, the head of the robot can be oriented utilizing the Euler angles provided by the Oculus Rift as can be appreciated in Fig. 4. These angles fulfill at a acceptable rate with the capabilities of the Inmoov head taking only the pitch and jaw components of the Euler angle. In addition to that the image resolution and the frame rate can be modified to ensure a fluid and vivid experience that does not fatigue the operator during operation. Altogether, the system is able to map the orientation of the robot arms, reacting to actions of opening and closing the hands by setting a threshold in the fingers orientation. The head is operated with the VR headset where the Euler angles are directly mapped and scaled to match with the two robot's head actuators. The mouth, also can be set to work synchronously with the a text-to-speech node to bring a more realistic experience to the robot collaborator. The system does not have any type of response for the joints placed on the heaps of the robot.

IV. Results

In order to validate how the motion mapping problem is addressed for this particular case, some tests were made under different scenarios using MATLAB and ROS. This tests complement the acceptable results obtained by the system in real situations, as shown in Fig. 1. On these tests, different left arm trajectories composed of a time window of 100 Euler angles for each joint, at a sample rate of 100 Hz were generated and then processed. In all tests, constant angular velocities from the operator's joints were assumed as well as the robot workspace's constraints. In this way, a visual resemblance of the animation generated by the MoCap suit, alongside with the virtual robot controlled by the Motion Mapping node, is shown in Fig. 6. This representation is shown using the ROS Vizualization tool (RViz).

Since the main issue relies on mapping the motion captured from the operator's arms to the robot's arms and because the rotation angles (θ_4 and θ_5) involved with the motion from the elbow to the wrist can directly mapped as shown in the Equation (2). The Fig. 7(a) shows a scenario where the left robot's elbow tracks the operators' motion with almost a perfect match, this test was performed using a trajectory composed of Euler angles on the ZYX configuration, from the rest position to the elbow's orientation denoted by $\left[\frac{3\pi}{4}, 0, 0\right]$. On the other hand, the Fig. 7(b) describes a movement where the shoulder displacement of the robot makes it impossible to reach the same position that the MoCap system throughout the trajectory, however, it always follows the same orientation of the operator's elbow. This test was performed using $\text{ZYX} = \left[\frac{\pi}{2}, \frac{\pi}{4}, \frac{\pi}{6}\right]$ as final orientation of one of the generated trajectories, also starting from the rest position. Hence, the resulting generalized coordinates for this particular test are represented in a subset of $\mathbf{q}$, described as $\hat{\mathbf{q}} = [\theta_1, \theta_2, \theta_3]^T \approx [1.1832, 0.3614, 0.8571]^T$. Finally, as can be appreciated in Fig. 8, the position error reach moderate values about $[x, y, z]^T \approx [0.038, 0.022, 0.071]^T$ meters, as shown in the Fig 7.(b), scenario where tracks the trajectory in terms in position is not achievable for the robot.

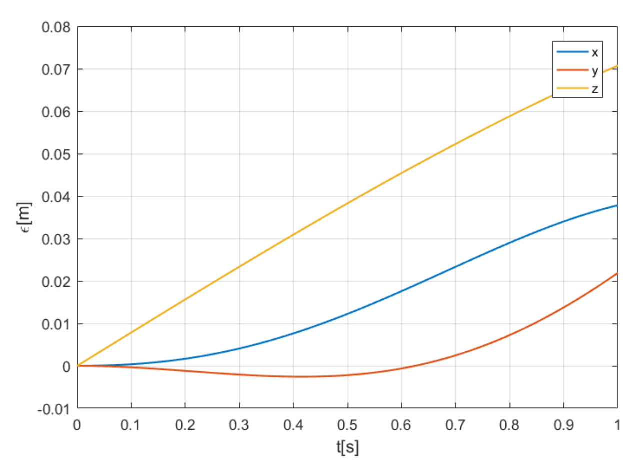

Fig. 8. Position error of the left elbow of Leonardo.

Fig. 9. Robot Head controlled using the Oculus Rift DK2.

Besides, satisfactory results were obtained using the Oculus Rift DK2 to maps motions from the operator's head, as shown in Fig. 9. The only rotational motion constraint is the absence of the intrinsic rotation about the x axis in the robot's neck (roll angle) due the amount of DOAs from the robot's head configuration, as mentioned in the Subsection II.C and as is illustrated in Fig. 4.

V. Conclusions

Since, the head of the robot provides an immersive VR experience to the operator giving him different an innovative ways to interact with an audience, inasmuch as the robot's surroundings are presented through stereo images and sound. Therefore it is interesting for future works, to measure both user and audience experience after interacting with the system. In this particular case, the concept of using humanoid robots to involve people on the tech fields, has been applied on different types of audience, reaching high acceptance and interest. Moreover, the arms' orientation are replicated at a successful level, that in some cases, could achieve the exact orientation of the operator's arm, depending on the proper implementation of a closed loop control system and on the actuators' constraints. The position error may vary, because of the displacement l in the robot's shoulder configuration. In some of the tests, the suit loses precision relatively faster than the VR headset and is necessary to re calibrate it to continue working properly. Finally, the system has been tested during long iterations of a maximum of 3 hours without showing fatigue or misalignment.

References

[1] N. S. Pollard, J. K. Hodgins, M. J. Riley, and C. G. Atkeson, "Adapting human motion for the control of a humanoid robot," in *Proceedings 2002 IEEE international conference on robotics and automation (Cat. No. 02CH37292)*, vol. 2. IEEE, 2002, pp. 1390–1397.

[2] G. H. Ballantyne, "Robotic surgery, telerobotic surgery, telepresence, and telementoring," *Surgical Endoscopy and Other Interventional Techniques*, vol. 16, no. 10, pp. 1389–1402, 2002.

[3] D. Gong, J. Yu, and G. Zuo, "Motion mapping for the heterogeneous master-slave teleoperation robot using unit dual quaternions," in *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*. IEEE, 2017, pp. 1194–1199.

[4] K. Theofilis, J. Orlosky, Y. Nagai, and K. Kiyokawa, "Panoramic view reconstruction for stereoscopic teleoperation of a humanoid robot," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 242–248.

[5] U. A. Chattha and M. A. Shah, "Survey on causes of motion sickness in virtual reality," in *2018 24th International Conference on Automation and Computing (ICAC)*. IEEE, 2018, pp. 1–5.

[6] K. Rohr, "Towards model-based recognition of human movements in image sequences," *CVGIP: Image understanding*, vol. 59, no. 1, pp. 94–115, 1994.

[7] D. Fitzgerald, J. Foody, D. Kelly, T. Ward, C. Markham, J. McDonald, and B. Caulfield, "Development of a wearable motion capture suit and virtual reality biofeedback system for the instruction and analysis of sports rehabilitation exercises," in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2007, pp. 4870–4874.

[8] J. R. Whitson, "The new spirit of capitalism in the game industry," *Television & New Media*, p. 1527476419851086, 2019.

[9] D. N. Aratuo and X. L. Etienne, "Industry level analysis of tourism-economic growth in the united states," *Tourism Management*, vol. 70, pp. 333–340, 2019.

[10] P. Morasso, M. Casadio, P. Giannoni, L. Masia, V. Sanguineti, V. Squeri, and E. Vergaro, "Desirable features of a "humanoid" robot-therapist," in *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2009, pp. 2418–2421.

[11] A. Guneysu, B. Arnrich, and C. Ersoy, "Children's rehabilitation with humanoid robots and wearable inertial measurement units," in *Proceedings of the 9th International Conference on Pervasive Computing Technologies for Healthcare*. ICST (Institute for Computer Sciences, Social-Informatics and ..., 2015, pp. 249–252.

[12] N. Miller, O. C. Jenkins, M. Kallmann, and M. J. Mataric, "Motion capture from inertial sensing for untethered humanoid teleoperation," in *4th IEEE/RAS International Conference on Humanoid Robots, 2004.*, vol. 2. IEEE, 2004, pp. 547–565.

[13] H. Yi, C. Knabe, T. Pesek, and D. W. Hong, "Experiential learning in the development of a darwin-hp humanoid educational robot," *Journal of Intelligent & Robotic Systems*, vol. 81, no. 1, pp. 41–49, 2016.

[14] P. Rezeck, F. Cadar, J. Soares, B. Frade, L. Pinto, H. Azpurua, D. G. Macharet, L. Chaimowicz, G. Freitas, and M. F. M. Campos, "An immersion enhancing robotic head-like device for teleoperation," in *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*. IEEE, 2018, pp. 164–169.

[15] G. Langevin, "Inmoov open-source 3d printed life-size robot," *pp. URL: http://inmoov.fr, License: http://creativecommons. org/licenses/by–nc/3.0/legalcode*, 2014.

[16] M. Sakashita, T. Minagawa, A. Koike, I. Suzuki, K. Kawahara, and Y. Ochiai, "You as a puppet: Evaluation of telepresence user interface for puppetry," in *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. ACM, 2017, pp. 217–228.

[17] F. Bazzano and F. Lamberti, "Human-robot interfaces for interactive receptionist systems and wayfinding applications," *Robotics*, vol. 7, no. 3, p. 56, 2018.

[18] R. Karunasenal, P. Sandarenu, M. Pinto, A. Athukorala, R. Rodrigo, and P. Jayasekara, "Devi: Open-source human-robot interface for interactive receptionist systems," 2019.

[19] M. Meredith, S. Maddock *et al.*, "Motion capture file formats explained," *Department of Computer Science, University of Sheffield*, vol. 211, pp. 241–244, 2001.

[20] S. M. LaValle, A. Yershova, M. Katsev, and M. Antonov, "Head tracking for the oculus rift," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 187–194.

[21] W. Xu, X. Li, W. Xu, L. Gong, Y. Huang, Z. Zhao, L. Zhao, B. Chen, H. Yang, L. Cao *et al.*, "Human-robot interaction oriented human-in-the-loop real-time motion imitation on a humanoid tri-co robot," in *2018 3rd International Conference on Advanced Robotics and Mechatronics (ICARM)*. IEEE, 2018, pp. 781–786.

[22] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *Journal of dynamic systems, measurement, and control*, vol. 108, no. 3, pp. 163–171, 1986.

[23] A. A. Maciejewski and J. M. Reagin, "A parallel algorithm and architecture for the control of kinematically redundant manipulators," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 4, pp. 405–414, 1994.