POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

AIRLAB
ARTIFICIAL INTELLIGENCE AND ROBOTICS LAB

EXECUTIVE SUMMARY OF THE THESIS

Non-Verbal Communication Through a Robotic Physical Avatar: A Study Using Minimal Sensor Information

LAUREA MAGISTRALE IN COMPUTER SCIENCE AND ENGINEERING - INGEGNERIA INFORMATICA

Author: GIUSEPPE EPIFANI

Advisor: PROF. ANDREA BONARINI

Co-advisor: FEDERICO ESPOSITI

Academic year: 2021-2022

---

1. Introduction

This thesis has been developed within a project that aims to study Embodiment using a physical avatar as a means of communication and interaction [3]. The project involves building a robot with non-humanoid features that can be remotely controlled and interact with people [2]. The research is divided into three areas: studying the robot's sensor information, physicality and interaction, and innovative control and feedback.

This thesis focuses on the first area and uses virtual reality (VR) to represent the robot's perceptions of its surroundings. The following chapters describe the steps taken in the project, such as choosing the information to pick up from the environment and developing software for data processing and transmission. A VR application has been developed in Unity which acts as an interface for the robot controller. Finally, the team conducted testing in the laboratory with subjects who filled out a questionnaire afterwards. The report of those results is presented in this work.

2. The Project

The project aims to investigate the effectiveness of controlling a robot and communicating with people through a virtual reality (VR) environment. The project involves conducting an in-person test, in which participants use a remotely controllable robot equipped with sensors and wheels to act as a physical avatar. The sensor readings are sent via Wi-Fi to a VR headset worn by the robot's controller and reprocessed to map the robot's signals in a virtual representation.

The project focuses on the detection of obstacles, humans, and dangerous situations, which are essential for environmental exploration and

interaction. A 360-degree Lidar scanner and an accelerometer sensor are used for obstacle detection, while a camera and a detection algorithm are used for human detection. The robot's sensors are connected to a Jetson Nano board, which controls them and sends data to the connected VR headset via the Local Network. The wheels of the robot are controlled by an Arduino and a Raspberry Pi.
The design of the VR environment involves translating each data type into a different element of the environment, with simple and clear representations to reduce delay and not introduce bias. The project also explores the benefits of using non-humanoid avatars in the VR environment to highlight diversity and representation and allow for creative and artistic expression of human presence.

3. The Choice of the Sensors

For the choice of the sensor we have chosen a bottom-up approach: experiment with a minimal amount of sensors, only those we thought were necessary to navigate and establish a communication. We performed an analysis on what kind of data we needed for those functions.
We have identified three fundamental elements for environmental exploration and interaction:

- The detection of obstacles, in order to have spacial awareness and be able to move around safely.
- The detection of humans, in order to enable the interaction.
- The detection of "danger", in order to avoid damaging the robot and hurting others.

Regarding the detection of obstacles, we decided to use a 360-degree Lidar scanner, in order to have a high-frequency omnidirectional reading of the environment.
We decided to pair the Lidar scanner with an accelerometer sensor in order to detect sudden changes of acceleration of the robot, possibly corresponding to crashes into obstacles.
For the detection of humans the today's obvious choice is to use a camera paired with a detection algorithm. For the objective of this project, we did not want to give the controller a clear view of the camera readings, but instead use a reprocessing of the algorithm output. By using non-humanoid avatars users have to actively interpret and understand the non-human shapes they encounter, providing an opportunity to highlight diversity and representation, allowing the exploration of different cultural, social, and physical traits in a way that humanoid avatars may not be able to do.

3.1. Other Hardware

The hardware used on the robot include the Nvidia Jetson Nano board, which provides the main computational power with a Nvidia Maxwell GPU with 128 CUDA cores, a quad-core ARM Cortex-A57 CPU, 4GB of LPDDR4 RAM, and a 16GB eMMC storage module. It also supports a range of AI frameworks and tools, and can be programmed in popular languages such as Python and C++. The operating system adopted is Ubuntu Linux.
All the sensors on the robot are connected to the Jetson Nano. The LiPo battery used to power the Jetson Nano is a rechargeable lithium polymer battery with a nominal voltage of 7.4V. A voltage regulator is used to convert the higher voltage of the battery to the required 5V power supply for the Jetson Nano.
The sensors and the Jetson Nano are mounted on a base with three omni-directional wheels (Triskarino - Figure 1). The wheels are controlled by a Raspberry Pi board connected to a wireless joystick.
Finally, the Oculus Quest 2 VR headset is used for the remote control of the robot in the experiment. It provides an immersive VR experience with high-quality visuals and intuitive controls.

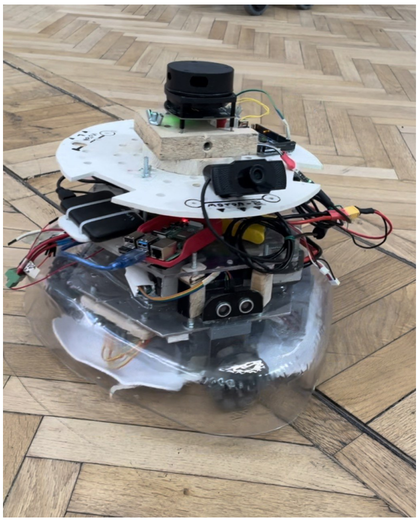

Figure 1: The final version of the robot

4. The Software

To control and process data from the sensors, a Python software application has been developed on the Jetson Nano. The software application comprises several modules, each responsible for a specific task. These modules include the LIDAR scanner module, the camera module, the MPU6050 sensor module, and the communication module.
The LIDAR sensor is controlled using the "rpLIDAR" library. The camera module not only captures live video feed, but also runs a pose detection algorithm to detect the location of a person near the robot. This is implemented using the "OpenCV" and "posenet" libraries. We chose to use a pose detection algorithm so that we could also detect individual body parts and get more information about the person's movement, rather than a simple bounding box detection. The data from the MPU6050 sensor connected to the Arduino Mega are acquired using the "pyserial" library in Python.
Regarding the connection between the robot and the VR headset, a networking solution has been developed. TCP protocol was used for reliable data transfer, while UDP protocol was utilized for faster transmission of non-critical data. The networking code was optimized for speed and low latency, implemented with recovery mechanisms for disconnections, and transmitted data asynchronously in separate threads to allow for parallel processing of incoming and outgoing messages. To accommodate the diversity of messages exchanged over the network, a communication protocol was necessary, with each sent message having a 1-byte overhead representing the type of the message. The headset and robot shared a dictionary of message types called "Keys," and listeners on both sides sorted and directed messages to the corresponding modules.

5. Design of the VR Environment

Regarding the design of the VR environment, we decided to develop different versions of the visualizations, in order to detect whether changes affect the user experience.
For the representation of the environment, we decided to stay as close as possible to a direct translation of the Lidar readings. We represented the distances detected from the sensor as walls in the environment, in two versions: a version in which the walls' height corresponds to the real-world height, and so appears as high as the VR user's head, and a version in which the walls appear much higher.

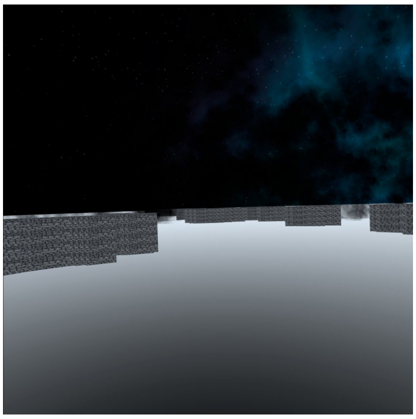

Figure 2: The environment with the low walls

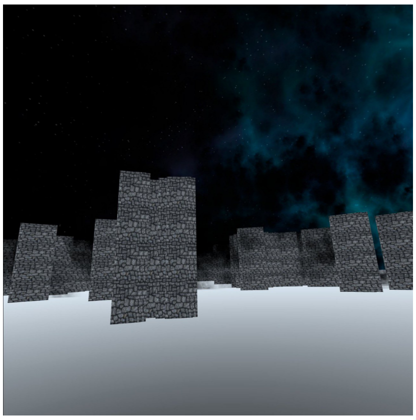

Figure 3: The environment with the high walls

We used the accelerometer readings to extract a single piece of information called "bump," calculated by taking the square of the instantaneous acceleration in the X- and Y-axes and checking if they exceeded a threshold calibrated by bumping into walls and objects. The bump was represented by turning the entire field of view red for an instant, chosen for its instinctive association with danger and the sense of priority it gives the user. The representation was found to be effective and intuitive in tests, with the controller moving the robot in the opposite direction after a bump.

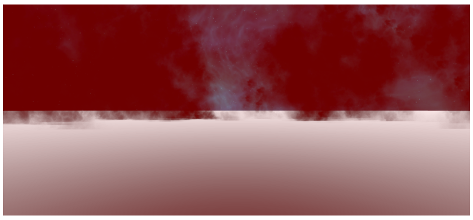

Figure 4: The environment during a bump

Regarding the representation of the detected human, we designed three different avatars, each fundamentally different from the others.
The first avatar is the Triangle. It is the closest representation to a human figure, as the vertices of the triangle correspond to the head and hands points of the detected person. There is minimal data reprocessing, as the keypoints from the pose detection algorithm are used directly in the virtual environment. This avatar was inspired by the human representation used in the research [Re]Moving bodies [1].

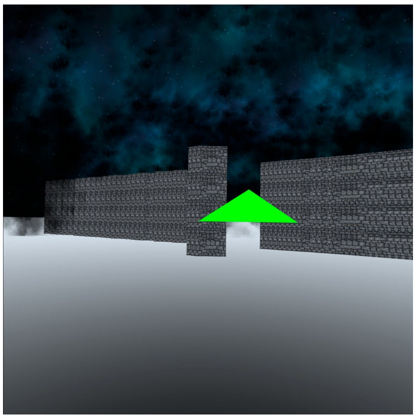

Figure 5: The Triangle avatar

The Entity representation is a creature we designed, inspired by the shape of a tubular coral species. It consists of a main body, some tentacles on top and an eye in the centre of the body. We chose this shape to study the reaction of people being in front of a non-anthropomorphic being, but with a physicality compatible with that of a human. This representation is the least abstract, and is controlled entirely by parameters and heuristics derived from the pose detection algorithm points.

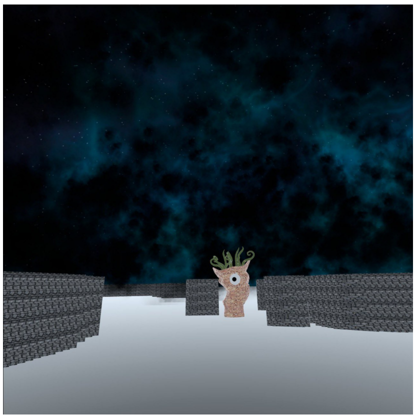

Figure 6: The Entity avatar

The Wave representation is the most abstract of the three. It consists of a large line in the sky. When there is no human detected, the line is flat and parallel to the horizontal plane. The line can move by waving following a sine-like function when a person is detected. With this design we wanted to focus more on an environmental representation, rather than on a physical one.

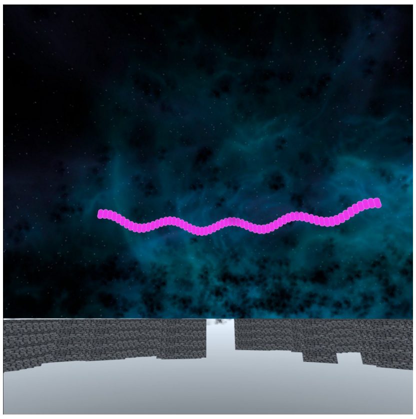

Figure 7: The Entity avatar

6. The Experiments

Experiments were conducted over three days in the AIRLab at the Politecnico di Milano with one person at a time. Waiting participants were placed in a separate room to eliminate possible influence on the personal experience. During the three-minute experiment, the participant explored the environment for the first minute without intervention, and then a staff member was introduced in the second minute to improvise an interaction with the robot. After the experiment, participants completed a questionnaire with three sections: general questions, environment and navigation, and perceived interaction, using a 5-point Likert scale.

7. Conclusions

We have presented this thesis project, a telepresence system designed for exploring and interacting in a virtual environment. According to the trials done, for the controllers navigation was successful, but interaction was lacking due to non-human representation and minimal sensors. Improvements could be made by experimenting with other avatars and pairing cameras with microphones or capacitive surfaces. A robotic arm is being developed for non-verbal communication, as well as a new control system with tactile feedback, necessary for embodied exploration.

References

[1] Julien Laroche, Loup Vuarnesson, Alexandra Endaltseva, Joseph Dumit, and Asaf Bachrach. [re] moving bodies–a shared diminished reality installation for exploring relational movement. *Frontiers in Psychology*, page 2036, 2021.

[2] Giuseppe Riva, John Waterworth, and Dianne Murray. *Interacting with Presence: HCI and the Sense of Presence in Computer-mediated Environments*. Walter de Gruyter GmbH & Co KG, 2014.

[3] Alexander Toet, Irene A Kuling, Bouke N Krom, and Jan BF Van Erp. Toward enhanced teleoperation through embodiment. *Frontiers in Robotics and AI*, 7:14, 2020.