Concepts for operating ground based rescue robots using virtual reality

Bachelor thesis by Jingyi Jia
Date of submission: July 4, 2021

1. Review: Prof. Dr. Max Mühlhäuser
2. Review: Julius von Willich
Darmstadt

TECHNISCHE UNIVERSITÄT DARMSTADT

Computer Science Department

Telekooperation
Prof. Dr. Max Mühlhäuser

Erklärung zur Abschlussarbeit
gemäß §22 Abs. 7 und §23 Abs. 7 APB der TU Darmstadt

Hiermit versichere ich, Jingyi Jia, die vorliegende Bachelorarbeit ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Fall eines Plagiats (§38 Abs. 2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Bei der abgegebenen Thesis stimmen die schriftliche und die zur Archivierung eingereichte elektronische Fassung gemäß §23 Abs. 7 APB überein.

Bei einer Thesis des Fachbereichs Architektur entspricht die eingereichte elektronische Fassung dem vorgestellten Modell und den vorgelegten Plänen.

Darmstadt, 4. Juli 2021

Jingyi Jia

Contents

1 Abstract

2 Introduction

In recent years, natural disasters such as earthquakes, tsunamis and potential nuclear, chemical, biological and explosives have seriously threatened the safety of human life and property. While the number of various disasters has increased, their severity, diversity and complexity have also gradually increased. The 72h after a disaster is the golden rescue time, but the unstructured environment of the disaster site makes it difficult for rescuers to work quickly, efficiently and safely.

Rescue robots have the advantages of high mobility and handling breaking capacity, can work continuously to improve the efficiency of search and rescue, and can achieve the detection of graph, sound, gas and temperature within the ruins by carrying a variety of sensors, etc. Moreover, the robot rescue can assist or replace the rescuers to avoid the injuries caused by the secondary collapse and reduce the risk of rescuers. Therefore, rescue robots have become an important development direction.

In fact, rescue robots have been put to use in a number of disaster scenarios. The Center for Robot-Assisted Search and Rescue (CRASAR) used rescue robots for Urban Search and Rescue (USAR) task during the World Trade Center collapse in 2001 [1] and has employed rescue robots at multiple disaster sites in the years since to assist in finding survivors, inspecting buildings and scouting the site environment etc [2]. Anchor Diver III was utilized as underwater support to search for bodies drowned at sea after the 2011 Tohoku Earthquake and Tsunami [3].

Considering the training time and space constraints for rescuers [4], and the goal of efficiency and fluency collaboration [5], the appropriate human-robot interaction approach deserves to be investigated. Some of the existing human-computer interaction methods are Android software [6] [7], gesture recognition[8] [9] [10], facial voice recognition [11], adopting eye movements [12], Augmented Reality(AR)[13] and Virtual Reality(VR), etc.

Among them, VR has gained a lot of attention due to its immersion and the interaction method that can be changed virtually. VR is no longer a new word. With the development of technology in recent years, VR devices are gradually becoming more accessible to users. With the improvement of hardware devices, the new generation of VR headsets have higher resolution and wider field of view. And in terms of handle positioning, with the development of computer vision in the past few years, VR devices can now use only the four cameras mounted on the VR headset to achieve accurate spatial positioning, and support hand tracking, accurately capturing every movement of hand joints. While VR are often considered entertainment devices, VR brings more than that. It plays an important role in many fields such as entertainment, training, education and medical care.

The use of VR in human-computer collaboration also has the potential. In terms of reliability, VR is reliable as a novel alternative to human-robot interaction. The interaction tasks that users can accomplish with VR devices do not differ significantly from those using real operating systems[14]. In terms of user experience and operational efficiency, VR displays can provide users with stereo viewing cues, which makes collaborative human-robot interaction tasks in certain situations more efficient and performance better [15].

However, there remains a need to explore human-computer interaction patterns and improve the level of human-computer integration[16]. Intuitive and easy-to-use interaction patterns can enable the user to explore the environment as intentionally as possible and improve the efficiency of search and rescue. The appropriate interaction method should cause less mental and physical exhaustion, which also extends the length of an operation, making it less necessary for the user to frequently exit the VR environment for rest.

For this purpose, this paper presents a preliminary VR-based system for the simulation of ground rescue robots with four different modes of operation and corresponding test scenes imitating a post-disaster city. The test scene simulates a robot collaborating with Unity to construct a virtual 3D scene. The robot has a simulated LiDAR remote sensor, which makes the display of the scene dependent on the robot's movement. In order to find an interaction approach that is as intuitive and low mental fatigue as possible, a user study was executed after the development was completed.

Chapter 3 talks about some of the research involving the integration of VR and human-computer interaction. Chapter 4 provides details of the purposed system, including the techniques used for the different interaction modes and the structure of the test scenes. Chapter 5 will talk about the design and process of user study. Chapter 6 presents the results of the user study and analyzes the advantages and disadvantages of the different modes of operation and the directions for improvement. Finally, in Chapter 7, conclusions and future work are summarized.

3 Related Work

In this chapter, some research on the integration of VR and human-computer interaction will be discussed. The relevant literature and its contributions will be briefly presented. The topic of VR and human-computer integration is an open research with many kinds of focus perspectives.

Robotic manipulation platforms combined with virtual worlds have several application scenarios. It can be used, for example, to train operators or to collaborate directly with real robots. Elias Matsas et al. [17] provided a VR-based training system using hand recognition. Kinect cameras are used to capture the user's positions and motions, and virtual user models are constructed in the VR environment based on the collected data to operate robots as well as virtual objects, such as buttons. Users will learn how to operate the robot in a VR environment. The framework of VR purposed by Luis Pérez et al. [18] is applied to train operators to learn to control the robot. Since the environment does not need to change in real time, but rather needs to realistically recreate the factory scene, the VR scene here is not reconstructed in a way that it is captured and rendered in real time. Rather, a highly accurate 3D environment was reconstructed in advance using Blender after being captured with a 3D scanner.

Building 3D scenes in virtual worlds based on information collected by robots is also a research highlight. Wang et al. [16] were concerned with the visualization of the rescue robot and its surroundings in a virtual environment. The proposed human-robot interaction system uses incremental 3D-NDT map to render the robot's surroundings in real time. The user can view the robot's surroundings in a first-person view through the HTC-Vive and send control commands through the handle's arrow keys. The novel VR-based practical system provided by Patrick Stotko et al. [19] uses distributed systems to reconstruct 3D scene. The data collected by the robot is fiest transmitted to the client responsible for reconstructing the scene. After the client has constructed the 3d scene, the set of actively reconstructed visible voxel blocks is sent to the server responsible for communication, which has a robot-based live telepresence and teleoperation system. This server will then broadcast the data back to the client used by the operator, thus enabling an immersive visualization of the robot within the scene.

Others are more concerned about the manipulation of the robotic arm mounted on the robot. Moniri et al. [20] provided a VR-based operating model for the robotic arm. The user wearing a headset can see a simulated 3D scene at the robot's end and send pickup commands to the remote robot by clicking on the target object with the mouse. The system proposed by Ostanin et al. [21] is also worth mentioning. Although their proposed system for operating a robotic arm is based on mixed reality(MR), the article is highly relevant to this paper, considering the correlation of MR and VR and the proposed system detailing the combination of ROS and robotics. In their system, the ROS

Kinect was used as middleware and was responsible for communicating with the robot and the Unity side. The user can control the movement of the robot arm by selecting predefined options in the menu. In addition, the orbit and target points of the robot arm can be set by clicking on a hologram with a series of control points.

4 Implementation

In this chapter, the tools and techniques used in building this human-computer collaborative VR-based system are described. The focus will be on interaction techniques for different modes of operation. In addition, the setup of the robot and the construction of test scenes will also be covered in this chapter.

4.1 Overview

The main goal of this work is to design and implement a VR-based human-robot collaboration system with different methods of operating the robot in order to find out which method of operation is more suitable to be used to control the rescue robot. Further, it is to provide some basic insights for future development directions and to provide a general direction for finding an intuitive, easy-to-use and efficient operation method. Therefore, the proposed system was developed using Unity, including four modes of operation and a corresponding test environment for simulating post-disaster scenarios. In each operation mode, the user has a different method to control the robot. In addition, in order to better simulate the process by which the robot scans its surroundings and the computer side cumulatively gets a reconstructed 3D virtual scene, the test environment was implemented in such a way that the picture seen by the user depends on the robot's movement and the trajectory it travels through.

4.2 System Architecture

The proposed system runs on a computer with the Windows 10 operating system. This computer has been equipped with an Intel Core i7-8700K CPU, 32 GB RAM as well as a NVIDIA GTX 1080 GPU with 8 GB VRAM. HTC Vive is used as a VR device. It has a resolution of 1080×1200 per eye, resulting in a total resolution of 2160×1200 pixels, a refresh rate of 90 Hz, and a field of view of 110 degrees. It includes two motion controllers and uses two Lighthouses to track the position of the headset as well as the motion controllers.

Unity was chosen as the platform to develop the system. Unity is a widely used game engine with a Steam VR plugin [1], which allows developers to focus on the VR environment and interactive behaviors in programming, rather than specific controller buttons and headset positioning, making VR development much simpler. Another reason why Unity was chosen as a development platform was the potential for collaboration with the Robot Operating System (ROS), a frequently used operating system for robot simulation and manipulation, which is flexible, low-coupling, distributed, open source, and has a powerful and rich third-party feature set. In terms of collaboration between Unity and ROS, Siemens provides open source software libraries and tools in C# for communicating with ROS from .NET applications [2]. Combining ROS and Unity to develop a collaborative human-robot interaction platform proved to be feasible [22]. Since the focus of this paper is on human-robot interaction, collaboration and synchronization of ROS will not be explored in detail here.

4.3 Robot

To simulate the process of a robot using a LiDAR remote sensor to detect the real environment and synchronise it to Unity, a sphere collision body was set up on the robot. The robot will transform the Layers of the objects in the scene into visible Layers by collision detection and a trigger event (onTriggerEnter function). The robot's driving performance, such as the number of collisions, average speed, total distance, etc., will be recorded in each test. The detailed recorded information can be seen in Fig.4.1. The movement of the robot depends on the value of the signal that is updated in each mode. In addition, the robot's Gameobject has the NavMeshAgent [3] component, which supports the robot's navigation to the specified destination with automatic obstacle avoidance in the test scene. The virtual robot has three cameras. One of the cameras is a simulation of a surveillance camera mounted on the robot, which can see all the items in the scene, although the distant items are not yet detected by LiDAR. Two of these camera are set up in such a way that they can only see the area detected by the robot's LiDAR remote sensor. Each camera captures what it sees and modifies the bound image bound in real time. The four operation modes described later all use the camera viewport as a monitoring screen by rendering the camera viewport on UI canvas.

4.4 Interaction techniques

This system has 4 different approaches to control the robot. Each mode has its own distinctive features:

1. In Handle Mode the user will send control commands directly using the motion controller.

[1] https://assetstore.unity.com/packages/tools/integration/steamvr-plugin-32647
[2] https://github.com/siemens/ros-sharp
[3] https://docs.unity3d.com/ScriptReference/AI.NavMeshAgent.html

2. In Lab Mode a simulated lab is constructed in the VR environment and the user will use virtual buttons in the lab to control the rescue robot.

3. In Remote Mode the user can set the driving destination directly.

4. In UI Mode the user has a virtual menu and sends commands via rays from the motion controller.

In order to improve the reusability of the code and to facilitate the management of subsequent development, the classes that manage the interaction actions of each mode implement the same interface. A graphical representation of the system structure is given in the UML activity diagram in Fig.4.1.

4.4.1 Handle Mode

In this mode, the user is controlling the robot's movement directly through the motion controller in the right hand. The touch pad of the motion controller determines the direction of rotation of the robot. The user can control the robot's driving speed by pulling the Trigger button. Fig.4.2 shows how to get the values from the HTC motion controller. The robot rotation direction will read the value of the touchpad X-axis. The range of values is $[-1, 1]$. Forward speed reads the Trigger button passed in as a variable of type SteamVR_Action_Single, and the range of the variable is $[0, 1]$. With the right-hand menu button, the surveillance screen around the robot can be turned on or off. The monitor window can be adjusted to a suitable position by dragging and rotating it. In the literature dealing with VR and human-computer collaboration, many researchers have used a similar operational approach. Therefore, as a widely used, and in a sense default operation approach, this mode was designed and became one of the proposed operation modes.

4.4.2 Lab Mode

The original intention of designing this mode is that there is a part of the literature where the immersive human-robot collaborative framework are used to train operators how to operate the robot, avoiding risks and saving learning costs or directly as a platform for operating the robot [18][17]. Therefore, in this mode, a virtual laboratory environment is constructed, in which simulated buttons, controllers, and monitoring equipment are placed. The laboratory consists of two parts. The first part is the monitoring equipment: the monitoring screen is enlarged and placed at the front of the lab as a huge display. The second part is the operating console in the center of the laboratory, which can be moved by the user as desired. The user can use the buttons on the right side to lock the robot or let it walk forward automatically. In the middle of the console are two operating joysticks that determine the robot's forward motion and rotation respectively. The part that involves virtual joystick movement and button effects uses an open source github project VRtwix[4]. With the sliding stick on the left, the user can edit the speed of the robot's forward movement and rotation.

[4]https://github.com/rav3dev/vrtwix

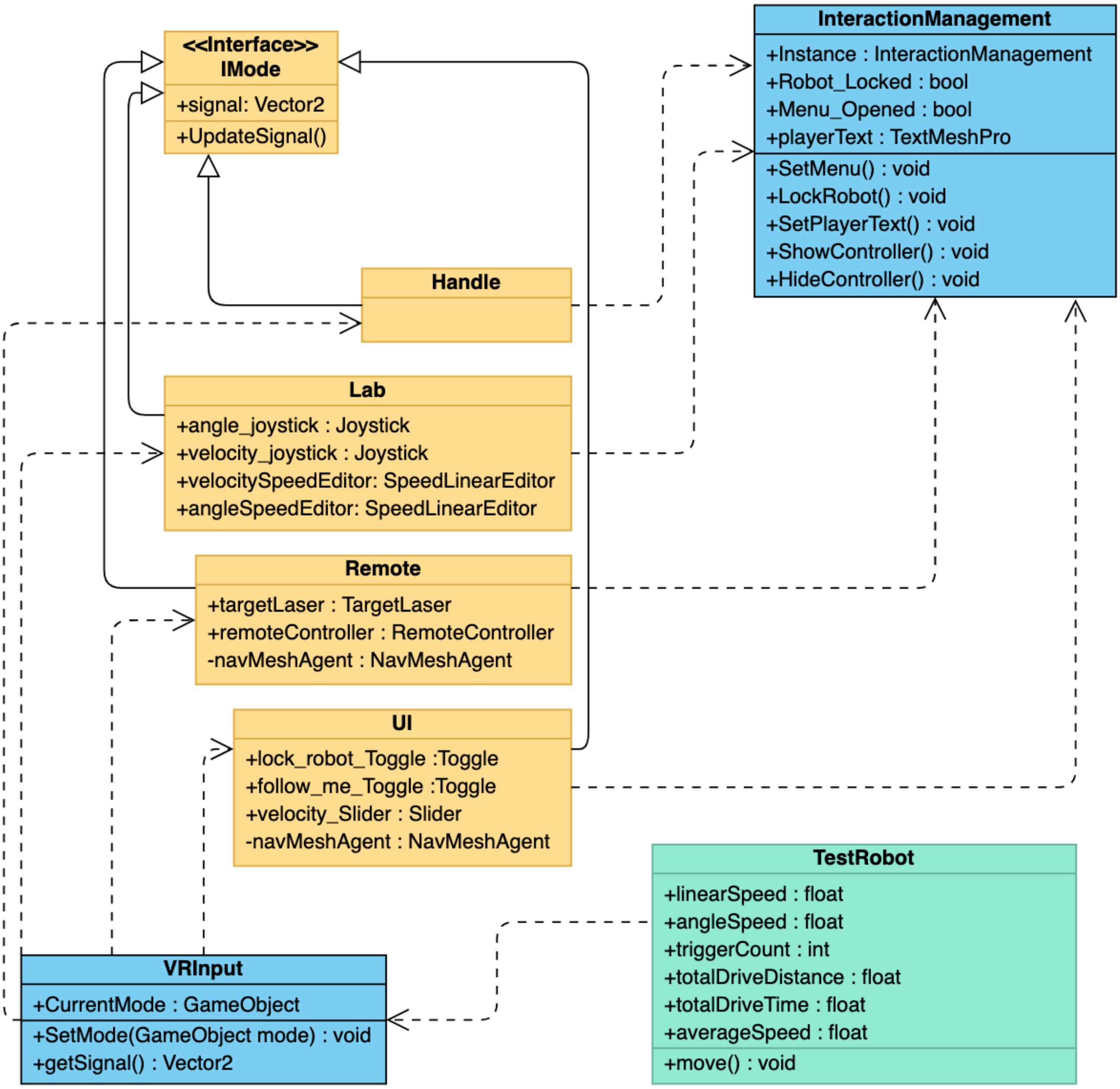

Figure 4.1: UML Class diagram for the main structure of the system

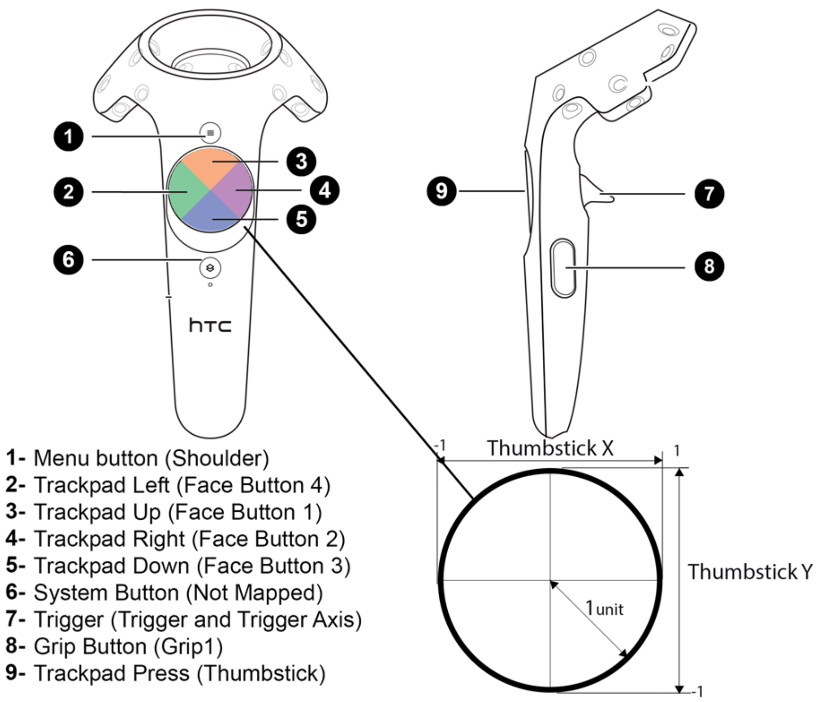

Figure 4.2: HTC handle illustration.

4.4.3 Remote Mode

In this mode, the user can set the driving target point directly or control the robot by picking up the remote control that is placed on the toolbar. The target point is set by the ray emitted by the right motion controller. This process is similar to setting a teleportation point. After the target point is set, a square representing the destination is shown in the scene, and the robot will automatically travel to the set destination. The entire driving process uses the NavMeshAgent component and is therefore capable of automatic obstacle avoidance. By clicking on the menu button, a movable toolbar is opened with a remote control and a monitoring device. The remote control is a safety precaution in case the automatic navigation fails to navigate to the target point properly. The user can adjust the direction of the robot's travel by using the remote control. The pickup and auto-release parts use the ItemPackage component available in the SteamVR plugin.

4.4.4 UI Mode

The virtual menu is also an interaction method that is often used in VR, so this mode is proposed. In this mode, the user must interact with the virtual menu using the ray emitted by the right motion controller. The virtual menu is set up with buttons for the direction of movement, speed controller, and buttons to open and close the monitor screen. In addition to this, an additional follow function is added to the menu, allowing the robot to follow the user's position in the virtual world. This is intended to let the user concentrate on observing the rendered VR environment. Also, having a real robot follow the user's location in the virtual world is a novel, unique human-machine integration mode in VR. The robot's automatic navigation uses the NavMeshAgent.

4.5 Test Scene

In order to simulate the use of rescue robots in disaster scenarios, the test scenes were built to mimic the post-disaster urban environment as much as possible. The POLYGON Apocalypse [5], available on the Unity Asset Store, is a low poly asset pack with a large number of models of buildings, streets, vehicles, etc. Using this resource pack as a base, additional collision bodies of the appropriate size were manually added to each building and obstacle after the pack was imported, which was needed to help track the robot's driving crash in subsequent tests.

Considering that there are four modes of operation to be tested, four scenes with similar complexity, similar composition of buildings but different road conditions and placement of buildings were constructed. The similarity in complexity of the scenes ensures that the difficulty of the four tests is basically identical. The different scene setups ensure that the scene information learned by the user after one test will not make him understand the next test scene and thus affect the accuracy of the test data.

The entire scene is initially invisible, and the visibility of each objects in the test scene is gradually updated as the robot drives along. Ten interactable sufferer characters were placed in each test scene. The place of placement can be next to the car, the house side and some other reasonable places.

[5]https://assetstore.unity.com/packages/3d/environments/urban/polygon-apocalypse-low-poly-3d-art-by-synty-154193

5 Evaluation of User Experience

This chapter describes the design and detailed process of the user evaluation. The purpose of this user study is to measure the impact of four different modes of operation on rescue efficiency, robot driving performance, and psychological and physiological stress and fatigue, etc. For this purpose, participants are asked to find victims in a test scene using different modes of operation and to answer questionnaires after the test corresponding to each mode of operation.

5.1 Study Design

The evaluation for each mode of operation consists of two main parts. The first part is the data recorded during the process of the participant driving the robot in the VR environment to find the victims. The recorded data includes information about the robot's collision and the speed of driving etc. The rescue of the victims was also considered as part of the evaluation. The Official NASA Task Load Index (TLX) was used to measure the participants subjective workload asessements. Additionally, participants were asked specific questions for each mode and were asked to select their favorite and least favorite operation mode. In order to reduce the influence of order effects on the testl results, the Balanced Latin Square was used when arranging the test order for the four operation modes.

5.2 Procedure

5.2.1 Demographics and Introduction

Before the beginning of the actual testing process, participants were informed of the purpose of the project, the broad process and the content of the data that would be collected. After filling in the basic demographics, the features of each of the four modes of operation and their rough usage were introduced verbally with a display of the buttons on the motion controllers.

5.2.2 Entering the world of VR

After the basic introduction part, participants would directly put on the VR headset and enter the VR environment to complete the rest of the tutorial. Considering that participants might not have experience with VR and that it would take time to learn how to operate the four different modes, the proposed system additionally sets up a practice pattern and places some models of victims in the practice scene. After entering the VR world, participants first needed to familiarize themselves with the opening and closing menu, as well as useing the motion controllers to try to teleport themselves, or raise themselves into mid-air. Finally participants were asked to interact with the victim model through virtual hands. After this series of general tutorials, participants were already generally familiar with the use of VR and how to move around in the VR world.

5.2.3 Practice and evaluation of modes

Given the different manipulation approaches for each mode, in order to avoid confusion between the different modes, participants would then take turns practicing and directly evaluating each mode immediately afterwards.

The sequence of modes to be tested is predetermined. The order effect is an important factor affecting the test results. If the order of the operational modes to be tested was the same for each participant, the psychological and physical exhaustion caused by the last operation mode would inevitably be greater. In order to minimize the influence of the order effect on the results of the test, the Balanced Latin Square with the size of four was used to arrange the test order of the four operation modes.

Participants automatically entered the practice scene corresponding to the relevant operation mode in the predefined order. After attempting to rescue 1-2 victim models and the participant indicated that he or she was familiar enough with this operation mode, the participant would enter the test scene. In the test scene, participants had to save as many victims as possible in a given time limit. Participants were required to move the robot around the test scene to explore the post-disaster city and to find and rescue victims. In this process, if the robot crashes with buildings, obstacles, etc., besides the collision information being recorded as test data, participants would also receive sound and vibration feedback. The test will automatically end when time runs out or when all the victims in the scene have been rescued. Participants were required to complete the evaluation questionnaire and the NASA evaluation form at the end of each test. This process was repeated in each mode of operation.

After all the tests were completed, participants were asked to compare the four operation modes and select the one they liked the most and the one they liked the least. In addition, participants could give their reasons for the choice and express their opinions as much as they wanted, such as suggestions for improvement or problems found during operation.

6 Results

7 Conclusion and future work

Bibliography

[1] J. Casper and R. R. Murphy. “Human-robot interactions during the robot-assisted urban search and rescue response at the World Trade Center”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 33.3 (2003), pp. 367–385. DOI: 10.1109/TSMCB.2003.811794.

[2] R. R. Murphy. “A decade of rescue robots”. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2012, pp. 5448–5449. ISBN: 2153-0866. DOI: 10.1109/IROS.2012.6386301.

[3] Ya-Wen Huang et al. “Operation of underwater rescue robot anchor diver III during the 2011 Tohoku Earthquake and Tsunami”. In: *OCEANS’11 MTS/IEEE KONA*. 2011, pp. 1–6. ISBN: 0197-7385. DOI: 10.23919/OCEANS.2011.6107198.

[4] R. R. Murphy. “Human-robot interaction in rescue robotics”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 34.2 (2004), pp. 138–153. DOI: 10.1109/TSMCC.2004.826267.

[5] Guy Hoffman and Cynthia Breazeal. “Effects of Anticipatory Action on Human-Robot Teamwork Efficiency, Fluency, and Perception of Team”. In: *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction*. HRI ’07. Arlington, Virginia, USA: Association for Computing Machinery, 2007, pp. 1–8. ISBN: 9781595936172. DOI: 10.1145/1228716.1228718. URL: https://doi.org/10.1145/1228716.1228718.

[6] S. Sarkar et al. “Earthquake rescue robot: A purview to life”. In: *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)*. 2017, pp. 1–7. DOI: 10.1109/ICECCT.2017.8118044.

[7] F. Faisal and S. A. Hossain. “DOORMOR: A Functional Prototype of a Manual Search and Rescue Robot”. In: *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*. 2019, pp. 1–6. DOI: 10.1109/ICASERT.2019.8934515.

[8] Patrick Sousa et al. *Human-Robot Interaction Based on Gestures for Service Robots*. Vol. 27. Oct. 2017, pp. 700–709. ISBN: 978-3-319-68194-8. DOI: 10.1007/978-3-319-68195-5{_}76.

[9] Alessandro Giusti et al. “Human-Swarm Interaction through Distributed Cooperative Gesture Recognition”. In: *Proceedings of the Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction*. HRI ’12. Boston, Massachusetts, USA: Association for Computing Machinery, 2012, pp. 401–402. ISBN: 9781450310635. DOI: 10.1145/2157689.2157818. URL: https://doi.org/10.1145/2157689.2157818.

[10] J. Nagi et al. "Human-swarm interaction using spatial gestures". In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2014, pp. 3834–3841. ISBN: 2153-0866. DOI: 10.1109/IROS.2014.6943101.

[11] S. Pourmehr et al. ""You two! Take off!": Creating, modifying and commanding groups of robots using face engagement and indirect speech in voice commands". In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2013, pp. 137–142. ISBN: 2153-0866. DOI: 10.1109/IROS.2013.6696344.

[12] J. Ma et al. "A Novel EOG/EEG Hybrid Human–Machine Interface Adopting Eye Movements and ERPs: Application to Robot Control". In: *IEEE Transactions on Biomedical Engineering* 62.3 (2015), pp. 876–889. DOI: 10.1109/TBME.2014.2369483.

[13] João Soares, Alberto Vale, and Rodrigo Ventura. "A Multi-purpose Rescue Vehicle and a human–robot interface architecture for remote assistance in ITER". In: *Fusion Engineering and Design* 98-99 (2015). Proceedings of the 28th Symposium On Fusion Technology (SOFT-28), pp. 1656–1659. ISSN: 0920-3796. DOI: https://doi.org/10.1016/j.fusengdes.2015.06.148. URL: https://www.sciencedirect.com/science/article/pii/S092037961530199X.

[14] V. Villani, B. Capelli, and L. Sabattini. "Use of Virtual Reality for the Evaluation of Human-Robot Interaction Systems in Complex Scenarios". In: *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. 2018, pp. 422–427. ISBN: 1944-9437. DOI: 10.1109/ROMAN.2018.8525738.

[15] O. Liu et al. "Understanding human-robot interaction in virtual reality". In: *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. 2017, pp. 751–757. ISBN: 1944-9437. DOI: 10.1109/ROMAN.2017.8172387.

[16] P. Wang et al. "A novel human-robot interaction system based on 3D mapping and virtual reality". In: *2017 Chinese Automation Congress (CAC)*. 2017, pp. 5888–5894. DOI: 10.1109/CAC.2017.8243836.

[17] Elias Matsas and George-Christopher Vosniakos. "Design of a virtual reality training system for human–robot collaboration in manufacturing tasks". In: *International Journal on Interactive Design and Manufacturing (IJIDeM)* 11.2 (2017), pp. 139–153. DOI: 10.1007/s12008-015-0259-2. URL: https://doi.org/10.1007/s12008-015-0259-2.

[18] Luis Pérez et al. "Industrial robot control and operator training using virtual reality interfaces". In: *Computers in Industry* 109 (2019), pp. 114–120. DOI: https://doi.org/10.1016/j.compind.2019.05.001. URL: https://www.sciencedirect.com/science/article/pii/S0166361518308546.

[19] Patrick Stotko et al. *A VR System for Immersive Teleoperation and Live Exploration with a Mobile Robot*. Nov. 2019. DOI: 10.1109/IROS40897.2019.8968598.

[20] M. M. Moniri et al. "Human Gaze and Focus-of-Attention in Dual Reality Human-Robot Collaboration". In: *2016 12th International Conference on Intelligent Environments (IE)*. 2016, pp. 238–241. ISBN: 2472-7571. DOI: 10.1109/IE.2016.54.

[21] M. Ostanin et al. “Human-robot interaction for robotic manipulator programming in Mixed Reality”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. 2020, pp. 2805–2811. ISBN: 2577-087X. DOI: 10.1109/ICRA40945.2020.9196965.

[22] D. Whitney et al. “ROS Reality: A Virtual Reality Framework Using Consumer-Grade Hardware for ROS-Enabled Robots”. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2018, pp. 1–9. ISBN: 2153-0866. DOI: 10.1109/IROS.2018.8593513.