Towards An Intuitive Virtual Reality Interface using Cable-Driven Parallel Robots for Space Exploration

[1a]Nathan Kassai
Department of Mechanical Engineering
University of Nevada, Las Vegas
Las Vegas, NV, USA
kassan2@unlv.nevada.edu

[1b]Zahir Abram Castrejon
Department of Mechanical Engineering
University of Nevada, Las Vegas
Las Vegas, NV, USA
castrz1@unlv.nevada.edu

[2]Dr. Paul Oh
IEEE Fellow
Department of Mechanical Engineering
University of Nevada, Las Vegas
Las Vegas, NV, USA
paul.oh@unlv.edu

***Abstract*—Space Exploration is a continuously flourishing field of research, as NASA has a plethora of ongoing missions to be achieved over the next few years. With the advent of many robotic platforms dedicated for space exploration such as NASA's Dragonfly, their Mars Perserverance Rover, and many more, it is evident that these types of robots will continue to play a key role. Despite their success, the limited man power for such specialized operators, reliability concerns with Unmanned Aerial Vehicles (UAVs or drones) in such harsh environments, and the limited battery life justify the consideration of different approaches. This paper presents work towards a suspended Cable-Driven Parallel Robot (CDPR), paired with an intuitive Virtual Reality interface designed for space exploration. Real-time 3D Point Cloud visualization grants the operator a greater sense of immersion, and can allow any operator to view the environment around the CDPR much more clearly than current interfaces. Along with the benefits of a CDPR, an immersive VR interface gives operators intuitive control through rigorous tasks.**

***Index Terms*—cable driven parallel robot, space exploration, virtual reality**

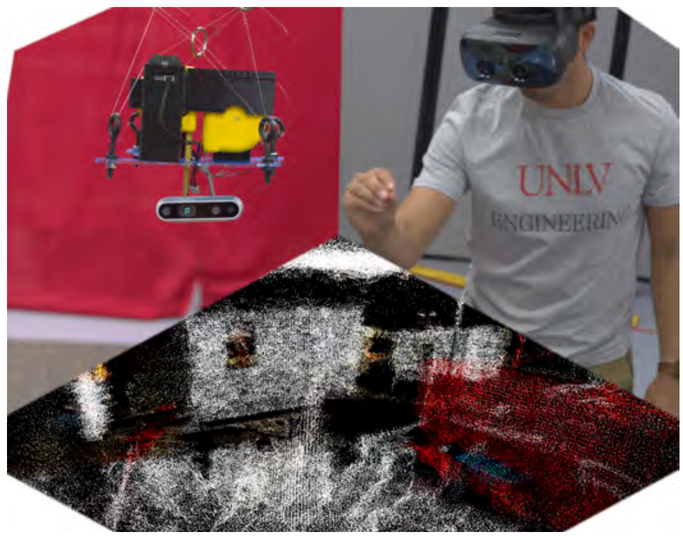

Figure 1: Conceptual Image

I. Introduction

Cable-Driven Parallel Robots (CDPRs) show promise in a variety of applications; namely land surveyal, disaster response, 3D printing, agricultural, etc. These platforms grant flexibility in their design to accomplish a large variety of tasks. For example, Narayanan *et. al.* demonstrated potential in a coconut farming CDPR capable of offloading the labor of collection and dehusking [1]. Moreover, Castrejon *et. al.* demonstrated a CDPR capable of large-scale warehouse monitoring [2]. Such flexibility is sought in Space Exploration as reliable methods to monitor specified areas on celestial bodies are still actively researched. Platforms most commonly seen in land surveyal and or exploration are Unmanned Aerial Vehicles (UAVs) and Unmanned Ground Vehicles (UGVs). Both provide distinct capabilities; namely an ability to survey land from a high point of view [3] [4], and or their simplistic, yet effective design respectively.

Despite these platforms demonstrating potential in optimizing space exploration, UGVs face limitations with challenging terrains such as steep cliffs and craters. Additionally, UAVs much like NASA's Ingenuity, while possessing enhanced mobility and high-speed land inspection, can fly for only 90 seconds. The reliability concerns with UAVs and UGVs in such harsh environments and limited battery life justify consideration for alternative approaches. In comparison to traditional systems, CDPRs offer advantages such as low inertia, adaptability, high payload-to-weight ratios, and spacious workspace volumes [5] [6]. Furthermore, CDPRs have unrestricted workspace movement, unlike UGVs limited vertically and UAVs constrained by power consumption.

Additionally, most of these systems have interfaces that are either difficult to control, and or lack a proper method of visualizing video feedback. For example, previous teleoperated

The work in this paper is partially by the Nevada NASA Space Grant AWD-02-00001955, the Nevada NASA EPSCOR AWD-02-00001965, Tesla Robotics in Construction AWD-02-00001843, and the INSPIRE UTC project is provided by the U.S. Department of Transportation, Office of the Assistant Secretary for Research and Technology (USDOT/OST-R) under Grant No. 69A3551747126 through INSPIRE University Transportation Center (http://inspire-utc.mst.edu) at Missouri University of Science and Technology. The views, opinions, findings and conclusions reflected in this publication are solely those of the authors and do not represent the official policy or position of the USDOT/OST-R, or any State or other entity.

systems rely only on a series of CCTV's from different perspectives [7] [8]. It is critical to design an interface encompassing full immersion, clarity of the robot's view, and intuitiveness in its design. A widely embraced approach is Virtual Reality (VR) as it simplifies task execution, providing immersive experiences for teleoperation and visualization users. [9] [10]. Therefore, a VR interface, in tandem with a CDPR, aims to highlight a potential robotic platform used for space exploration. Despite these benefits, there exist a few problems with the platform. These issues include mast anchoring uncertainties, Moon gantry delivery challenges, and difficulties in localizing and outrigging UGVs on Lunar terrain. This work highlights a CDPR framework towards mitigating these issues.

This paper is divided into five main sections: Section II explores related works. Section III details both the hardware, software, and Kinematics for the CDPR. Section IV details experiments with the full system. Finally, Section V draws conclusions regarding the proposed system as well as future work.

II. Related Work

Without major limitations in power consumption, CDPR's are free to move throughout their entire workspace volume, keeping the platform's end-effector suspended indefinitely. Due to this advantageous feature, Nurahami *et al.* proposed CDPRs for search and rescue operations [12]. Nurahami's system entailed limiting the amount of pillars and cables required to keep a CDPR suspended while retaining its navigational capability around challenging terrain. Their discussed platform demonstrates the minimum requirements to effectively allow a CDPR to maneuver and potentially lift objects around its environment.

As discussed prior, a need for versatile and robust robotic systems that incorporate a user-friendly Graphical User Interface (GUI) is desired. Latest teleoperational platforms follow a traditional paradigm of monitors for sensor visualization alongside joysticks or keyboards to update robot's poses. Whitney *et al.* demonstrated how these types of GUIs are outdated, and current mediums, such as VR, pose as preferable alternatives [11]. Using point clouds for visualization, Whitney designed three different remote interfaces, listing input and viewing method respectively, to control a Baxter robot. These interfaces included keyboard and monitor, positional hand tracking with a monitor, and positional hand tracking with VR Camera control. Whitney then had operators attempt to stack cups using their respective interface and monitored how fast each participant completed this task. Based on data from a NASA Task Load Index (NASA-TLX) and a System Usability Test, participants preferred the full VR interface significantly more than the other platforms. Additionally, participants found it easy to visualize Baxter's hand in proximity to cups, making this system highly intuitive. Furthermore, operators completed the cup stacking task in an average time of 53 seconds, in comparison to 153 seconds with traditional interfaces.

Figure 2: CDPR's Workspace

Visualization is a critical factor in teleoperational platforms which fall into the following two categories; 2D image streams or 3D Point Cloud generation. Participants prefer Point Cloud generation as they have a higher sense of situational awareness and higher success rates in high cognitive tasks [13]. Stotko *et al.* took this concept further and with the RGB-D data, performed a full scene reconstruction in VR [14]. Particpating operators found, after completing a System Usability Scale survey, that this visualization method was superior to 2D image streams. This superiority was attributed to the ease of assessing visual data of terrain around the robot and maneuvering difficult corners.

III. Methodology

A. Technical Design

For the defined workspace of 14'x14'x10' as seen in Fig. 2, four ROBOTIS Dynamixel PRO smart actuators are mounted behind each anchor point. Each motor is then used to adjust cable lengths such that desired speeds and positions are met. In the absence of external loads, these smart actuators reach speeds of 30 RPM and a total of 44.7 Nm of continuous torque. The motors are then mounted to a cable spool with a diameter of 150mm.

The end effector is mounted on an elliptical pendular suspension known as a Picavet, previously used for aerial kite photography [3]. This passive mechanism has four attachment points and a center ring, preventing unwanted twisting of the image. by constraining intersecting lines. As the CDPR base tilts, the Picavet provides a leveled plane without any additional actuation, stabilizing the image as shown by Fig. 3. Atop the Picavet is an NVIDIA Jetson Xavier NX, controlling

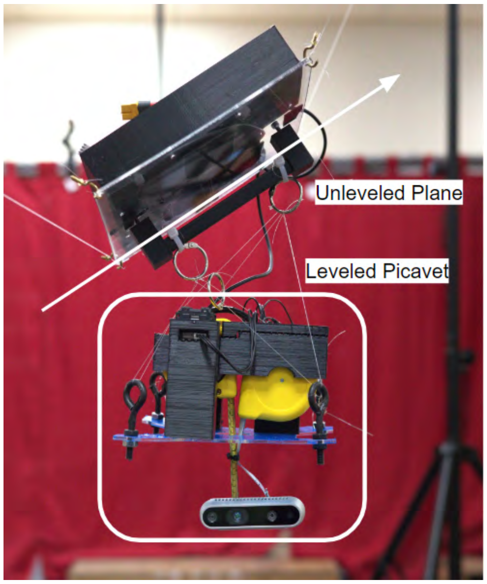

Figure 3: Closer look at Picavet Stabalizing the CDPR's End Effector

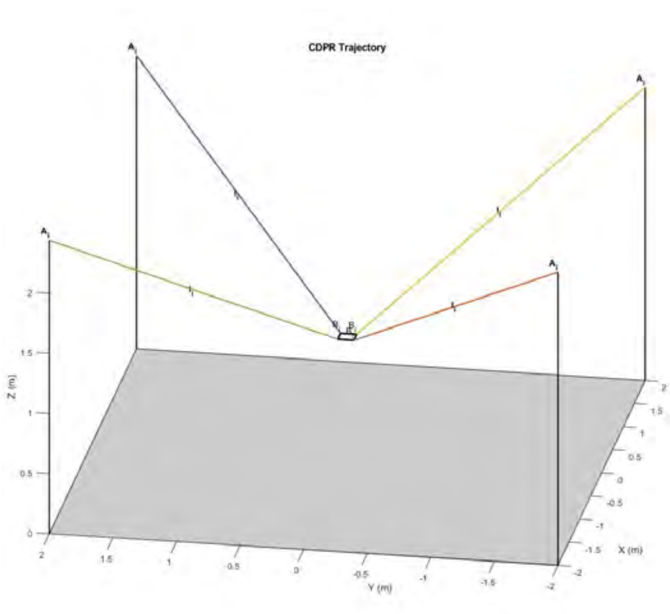

Figure 4: CDPR Simulation

Dynamixel AX-12 motors for the bi-stable mast. Connected to the Jetson is a RealSense D435i to capture point cloud data. To visualize the point cloud data and the interface, a Varjo XR3 Head Mounted Display (HMD) was provided. An additional benefit that this HMD provides is a built-in Ultraleap Hand Tracking sensor mounted in the front of the headset. This sensor records the operator's hand pose, which is then projected onto virtual hands in the Unity scene to interact with the interface.

A bi-stable telescopic mast is securely attatched to the Picavet. The design of this mechanism makes hard to reach places accessible for exploration and limits the chances of catastrophic collisions. As the end-effector moves through crevices, two coupled 16 foot long measuring tapes can extend and retract, keeping the CDPR and its cables static. Due to a tape measures ability to resist bending from its transverse curvature, this structure improves end-effector stability. [15]. Two Dynamixel AX-12 motors rotate in the same direction to reel these tapes together, thereby moving the D435i RealSense Camera vertically, and capturing additional points for the Point Cloud.

B. Kinematics

To provide position control of a CDPR with *m* DOFs and *n* cables, the length of each cable must correspond to a desired position. For a smooth motion, positive tension for each cable must be present; it is imperative that cable lengths change with respects to each DOF. The length of a cable is defined by anchor points in each corner as A_i and base points attached to the end-effector B_i. The motion from one point to another can be described as

$$\Delta L = J^{-1} \cdot \Delta P \tag{1}$$

where, ΔL is the change of cable lengths, J^{-1} is the inverse Jacobian matrix and ΔP is the change of position. We assume that cables are rigid and tension is positive for each cable.

1) Cable Anchors: For a reference frame, initial positions for the anchor points of a given workspace need to be defined. The cable anchors set at corners of the workspace are represented in a matrix to define its center and the lowest point of its end-effector. Dim_x represents the dimensions of the x-axis, Dim_y represents the dimension of the y-axis, and Dim_z as the height.

$$\text{Anchor Points} = \begin{pmatrix} -\frac{\text{Dim}_x}{2} & \frac{\text{Dim}_x}{2} & \frac{\text{Dim}_x}{2} & -\frac{\text{Dim}_x}{2} \\ -\frac{\text{Dim}_y}{2} & -\frac{\text{Dim}_y}{2} & \frac{\text{Dim}_y}{2} & \frac{\text{Dim}_y}{2} \\ \text{Dim}_z & \text{Dim}_z & \text{Dim}_z & \text{Dim}_z \end{pmatrix}$$

2) Initial Cable Lengths: The initial lengths of the cables can be calculated using the Euclidean distance formula:

$$L_i = \sqrt{(B_{i_x} - A_{i_x})^2 + (B_{i_y} - A_{i_y})^2 + (B_{i_z} - A_{i_z})^2}$$

where L_i is the length of the i-th cable.

3) Jacobian Matrix: For each of the four cables, the Jacobian matrix for the system is calculated as:

$$\begin{pmatrix} \frac{\partial x}{\partial L_1} & \frac{\partial x}{\partial L_2} & \frac{\partial x}{\partial L_3} & \frac{\partial x}{\partial L_4} \\ \frac{\partial y}{\partial L_1} & \frac{\partial y}{\partial L_2} & \frac{\partial y}{\partial L_3} & \frac{\partial y}{\partial L_4} \\ \frac{\partial z}{\partial L_1} & \frac{\partial z}{\partial L_2} & \frac{\partial z}{\partial L_3} & \frac{\partial z}{\partial L_4} \end{pmatrix}$$

The motion for the CDPR is then validated using Matlab (as seen in Fig. 4) and physical measurements of the cables were obtained from the Anchor Points and Base Points. Since the CDPR is suspended, gravity will naturally act as a 5-th cable pulling down in the z-axis. This downward pull maintains the cables positive tension but adds limitation within

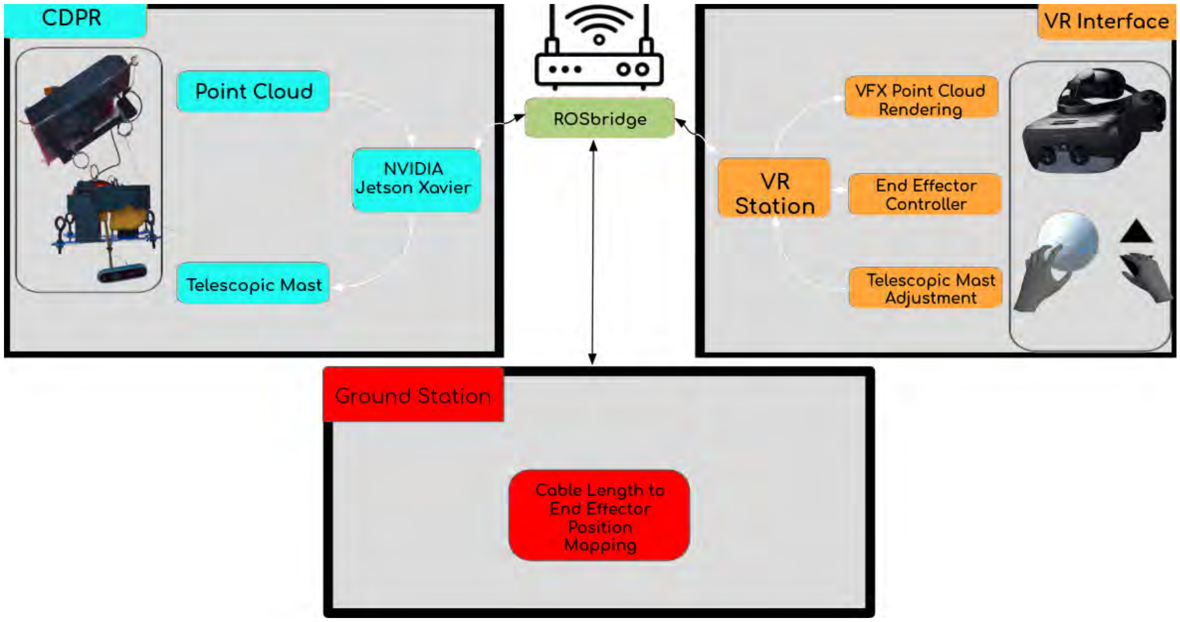

Figure 5: System overview diagram

obstacle avoidance. The ability to have 3D generated point cloud adds depth of field for the operator along with a bi-stable telescopic mast to assist against potential collisions. A z-axis adjustment independent of cable length change allows for surveying around fixed tall obstacles since there are physical constraints within the anchor points, cable lengths and overall workspace reconfigurabilty for obstacle avoidance.

C. Software

The keyword for this interface is *intuitive*, as to not overwhelm operators during missions. Hence, it is critical to limit the total amount of options available for operators. Each component of the interface must have a clear indication for its intended use. Furthermore, the most common medium for teleoperation is through VR. Operators find it much easier to complete tasks through this interface due to its intuitive controls and enhanced immersion, as previously mentioned. To construct an intuitive Virtual Reality interface, Unity, a popular game engine, was chosen for several reasons. These include seamless VR integration with various HMDs and compatibility with the Robot Operating System (ROS) via the ROS-Sharp Unity package. Additionally, it features an efficient particle generation tool capable of rendering millions of points without performance issues.

To reduce operator workload, the VR interface only contains three elements. These include Point Cloud visualization, end-effector position adjustment via a graspable sphere, and directional buttons to control the telescopic mast. This simplified mode of operation gives operators a sense of clarity as they are not overwhelmed with options.

In addition to the intuitiveness of the system's interface, a clear method of visualization of the CDPR's environment is critical for proper teleoperation. Initial tests with sending and rendering a live point cloud stream was found to be inefficient and poor due to the resolution of the Realsense camera. To remedy this issue, the Real-Time Appearance-Based Mapping (RTAB-Map) library, was used to generate a "Map Cloud" [16]. This algorithm employs an RGB-D SLAM approach, which incorporates loop closure detector; if an incoming image detected by the RGB-D sensor is from a new location, additional data is added to the map's overall graph. Furthermore, if the environment gets updated in the same location, RTAB-Map's algorithm will compare the RGB-D data in its graph at that position to the new data and make the update to the map cloud accordingly. With every accepted image read at 1 Hz, new points get added to the Map Cloud.

For our platform, we have three separate stations: VR station, Telescopic Mast and Realsense, and Ground Station. As aforementioned, ROS was the middleware selected for this project. The station in control of the Telescopic mast and the RealSense RTAB-Map generation was located on the Nvidia Jetson Xavier. The station in control of the translation of the CDPR was dubbed the Ground station. Both of these two stations had their respective ROS nodes running, as well as a ROSbridge[1] server that connects to the VR station's Unity scene. On the VR station's PC, ROS normally cannot connect directly, and would have to manually deserialize each message that gets received. Thanks to the ROS-sharp[2] package, this enables seamless connection between the ROS nodes on the ground station. In the Unity scene, there is a publisher node in charge of sending position data for the CDPR's end effector, and a subscriber that receives the incoming changes to the cloud map. An overview of the system can be found in Figure 5.

[1] http://wiki.ros.org/rosbridge_server
[2] https://github.com/siemens/ros-sharp

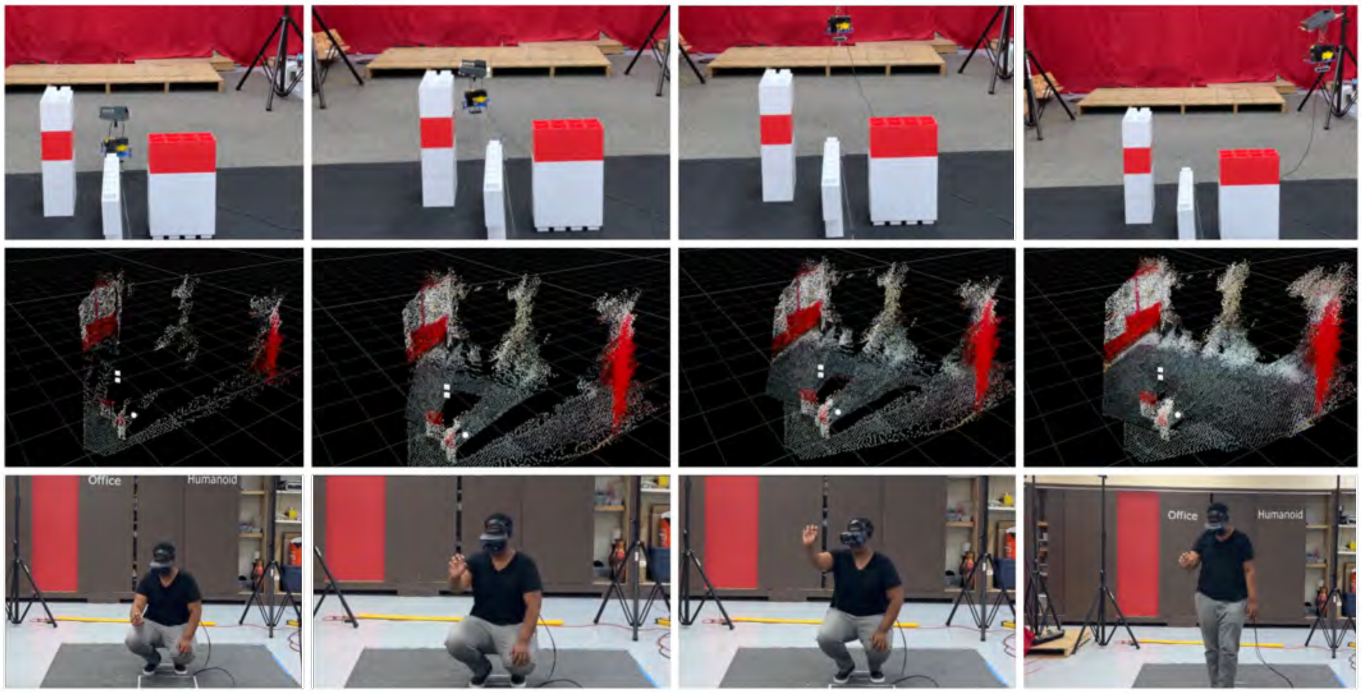

Figure 6: Operator demonstrating teleoperation via VR interface

Rendering point cloud data is notoriously difficult, as a colossal amount of data gets sent over the network with every scan that occurs. However, with RTAB-Map, this load gets drastically reduced, as only 5-10k points are sent every second, as opposed to millions of points per second. In order to render point cloud data received from ROS, it must first be deserialized and converted from its binary blob format, to a readable format; this is due to how ROS optimizes storing and sending its point cloud data. While many methods of visualizing point clouds in Unity exist, many share the same drawback; massive performance degradation due to millions of triangles having to be rendered for each point in a cloud. In order to optimize this rendering process, Unity's VFX Graph was used. The VFX Graph is a visual node-based interface which leverages the GPU to optimally handle rendering millions of points without hindering the performance of this Virtual Reality environment. Once point cloud data is received from ROS, this data is passed into two arrays: one to hold position, and another to hold RGB data of each point. This data then gets passed as variables to the Point Cloud VFX graph, to then assign an id, a position, and a color to the newly generated particle.

Initial tests found that points were insufficient for rendering, as particles were incredibly small, and limitations with the graphics library prevented adjustments to individual particle size. A viable alternative was to render each particle as a quad, then add an orient block to the VFX graph to have each rendered quads face the operator's camera. Alternatively, particles could have been rendered as cubes, but that was a waste of computational power, since faces that were not in direct view of the operator were still being rendered by Unity.

IV. Experiments

Intuitive control experiments for the CDPR were done within a 14'x14'x10' workspace and VR tracking space. To mimic the landscape of a martian environment, a handful of red boulders and small debris were placed within the CDPRs workspace. Prior to running experiments, the CDPR is required to undergo a calibration process in which its end-effector must be set at the center of its workspace and the lowest point with respects to the anchor points. The home position (0,0,0) is then defined by the calibration process and guarantees that each cable length corresponds to the desired position on the VR interface.

The VR interface is provided 3D point cloud data for spatial sensation as operators can plan appropriate trajectories of the end-effector. A 3D sphere is projected and represents the CDPR's current position and can be grasped by the operator from positional hand tracking to maneuver the end-effector around the workspace. Similar to holding a paint brush, the operator is able to capture the environment around the CDPR and safely navigate over and through debris. As the position is being updated, new data points are collected and rendered on the VR interface in real-time as seen in Figure 6. Despite well over four million points being rendered in the VR interface, the system had stayed well over 60 frames per second (FPS) all throughout teleoperation. To demonstrate the bi-stable telescopic mast, a switch for extension and retraction can be pressed on the VR interface without any changes in cable lengths. A recorded trial can be found at the following link demonstrating the system's capabilities: https://www.youtube.com/watch?v=fJRepSAi6s0.

V. Conclusion and Future Works

The works of this paper demonstrates an intuitive interface while utilizing a Cable-Driven Parallel Robot for space exploration. The CDPR allowed the operator to obtain visual 3D data and provided spacial sensation of an environment with MARS like props. Although the bridge between the Unity side and CDPR motions were reliable, an improvement for optimized 3D data points can be made for higher resolution and details within the environment. It was assumed that the system would not encounter any disturbances such as wind hence, a dynamic model accounting for such external disturbances would provide much help within this systems use case. Rotational movement of the camera was also not added to the current system, so operators were unable to view behind themselves. Adding an additional actuator to control the yaw of the RealSense would allow for a higher quality point cloud to be generated. Furthermore, alternative methods for point cloud rendering in Unity must be sought to gain a higher level of detail.

VI. Acknowledgment

The authors gratefully acknowledge the contributions of Thomas McNulty, Ethan Nowery, Akshay Dave, Baekseok Kim, Francis Palustre, and Santiago Ricoy.

References

[1] T. Narayanan, R. S. Vishnu, R. R. Bhavani and V. Vashista, "A cable driven parallel robot for coconut farm," 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, 2017, pp. 864-870

[2] A. Zargari, Z. A. Castrejon, D. Kim and P. Y. Oh, "Cable-Driven Parallel Robot for Warehouse Monitoring Tasks," 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 2023, pp. 1085-1090

[3] P. Y. Oh and W. E. Green, "Mechatronic kite and camera rig to rapidly acquire, Process, and distribute aerial images," in IEEE/ASME Transactions on Mechatronics, vol. 9, no. 4, pp. 671-678, Dec. 2004

[4] Green, WE, Oh, PY, and Yoon, S. "An Acquisition and Distribution System for Situational Awareness." Proceedings of the ASME 2003 International Mechanical Engineering Congress and Exposition. Dynamic Systems and Control, Volumes 1 and 2. Washington, DC, USA. November 15–21, 2003. pp. 1341-1346.

[5] D. Bury, J. -B. Izard, M. Gouttefarde and F. Lamiraux, "Continuous Collision Detection for a Robotic Arm Mounted on a Cable-Driven Parallel Robot," 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 2019, pp. 8097-8102

[6] Z. Zake, F. Chaumette, N. Pedemonte and S. Caro, "Visual Servoing of Cable-Driven Parallel Robots with Tension Management," 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 2021, pp. 6861-6867

[7] DeJong, B.P., Colgate, J.E., Peshkin, M.A. (2011), "Mental Transformations in Human-Robot Interaction", In: Wang, X. (eds) Mixed Reality and Human-Robot Interaction. Intelligent Systems, Control and Automation: Science and Engineering, vol 1010. Springer, Dordrecht.

[8] J. Mackro, "Study of design and control of remote manipulators". part 4: Experiments in video camera positioning with regard to remote manipulation, 1973.

[9] J. Y. C. Chen, E. C. Haas and M. J. Barnes, "Human Performance Issues and User Interface Design for Teleoperated Robots," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 37, no. 6, pp. 1231-1245, Nov. 2007

[10] J. I. Lipton, A. J. Fay and D. Rus, "Baxter's Homunculus: Virtual Reality Spaces for Teleoperation in Manufacturing," in IEEE Robotics and Automation Letters, vol. 3, no. 1, pp. 179-186, Jan. 2018

[11] Whitney, D., Rosen, E., Phillips, E., Konidaris, G., Tellex, S. (2020). "Comparing Robot Grasping Teleoperation Across Desktop and Virtual Reality with ROS Reality". In: Amato, N., Hager, G., Thomas, S., Torres-Torriti, M. (eds) Robotics Research. Springer Proceedings in Advanced Robotics, vol 10. Springer, Cham.

[12] L. Nurahmi, B. Pramujati, S. Caro and Jeffrey, "Dimension synthesis of suspended eight cables-driven parallel robot for search-and-rescue operation," 2017 International Conference on Advanced Mechatronics, Intelligent Manufacture, and Industrial Automation (ICAMIMIA), Surabaya, Indonesia, 2017, pp. 237-241, doi: 10.1109/ICAMIMIA.2017.8387594.

[13] Wonsick, M., Padir, T. (2020). "A systematic review of virtual reality interfaces for controlling and interacting with robots". Applied Sciences, 10(24), 1-17.

[14] P. Stotko et al., "A VR System for Immersive Teleoperation and Live Exploration with a Mobile Robot," 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 2019, pp. 3630-3637, doi: 10.1109/IROS40897.2019.8968598.

[15] O. G. Osele, A. M. Okamura and B. H. Do, "A Lightweight, High-Extension, Planar 3-Degree-of-Freedom Manipulator Using Pinched Bistable Tapes," 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 2022, pp. 1190-1196, doi: 10.1109/ICRA46639.2022.9811976.

[16] Labbé, M., and Michaud, F. (2019). "RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation". Journal of Field Robotics, 36(2), 416-446.