Improving teleoperation robots performance by eliminating view limit using 360 camera and enhancing the immersive experience utilizing VR headset

S Wibowo*, I Siradjuddin, F Ronilaya, and M N Hidayat
Department of Electrical Engineering, State Polytechnic of Malang, Jl. Soekarno Hatta No. 9, Malang, Indonesia

*sapto.wibowo@polinema.ac.id

Abstract. A teleoperated robotic system offers a rapid deployment for performing tasks in dangerous environments. An effective teleoperation requires a proper feedback system such as a vision system using cameras. Despite its effectiveness of using camera, the limited field of view (FoV) become one of the main burdens of the operation. To deal with this problem, some researchers used wider-view camera or multiple cameras. In contrast, this research proposed to achieve a better teleoperation by eliminating FoV limit by using 360 camera. This research also aimed to improve the immersive experience by utilizing virtual reality (VR) headset. The proposed framework then developed to accommodate all the requirements and then evaluated by some numerous trials by some participants. An AprilTag marker was placed on the robot and tracked by a ceiling mounted camera to record the trajectory data. The obtained trajectories then measured its performance by using a standard performance metrics. The experiments demonstrated that the proposed method achieved lower collision area, closer to desired ideal path and smoother than using a standard camera as the feedback system. However, as the 360-video streaming induced higher latency, the proposed approach was slower. The measurements justified that the proposed approach achieved a better teleoperation especially for mission critical tasks.

1. Introduction
Often, some works need to be done in dangerous environments, such as inspecting exploded nuclear power plant, getting some samples in viruses’ highly-contaminated areas or other dangerous situations. Sending human for such tasks is not a good option and the better alternative is to use robots to do the job. Full autonomous robots are an ideal solution for performing these tasks since autonomous robots capable to operates without the need of human [1]. To operate fully autonomous, the robots have to be able to recognise obstacle, know passable routes, planning the motion paths, create surrounding maps, localize itself within the environment, explore the environment efficiently and manipulate the objects as instructed. Those capabilities require complex algorithms such as SLAM, path planning, motion planning, object recognition, pose estimation, etc, which is non-trivial [2]. Hence, as those algorithms best work on a particular situation, there is no general-purpose autonomous algorithm that can be used for any environments without some significant adjustments. Different environments require non-trivial and time-consuming algorithm adjustments which make it difficult to implement promptly in different
complex missions. According to Mahmoud Zadeh et al, the full autonomy is still hard to realize for complex missions [3].

A considered approach for rapid deployment of robots on a dangerous, yet complex missions is to use human teleoperated robotic system. In this case, the robots are operated by human operators in safe area, which might miles apart from the robots. By inducing a human experience, the system takes the benefit of their intelligence that capable to promptly adapt arbitrary situations and perform complex tasks [4]. An effective teleoperation requires a proper feedback system such as LIDAR and vision system. Since images consist rich information that can be easily interpreted by human to build situational awareness, a vision system using camera(s) become a popular feedback system for teleoperated robotic systems. In this case, the operator observes the video streaming to perform the tele operation. However, despite its effectiveness of using camera, the limited field of view (FoV) of the camera become one of the main constrains of the tele operation. The limited FoV decrease the situational awareness and affect the performance of the operation [5].

To deal with this problem, some researchers used wider view or multiple cameras [5–9]. In contrast with those approaches, this research proposed to achieve a better teleoperation by eliminating FoV by using 360 camera. The 360 cameras practically can be seen to have no limited field of view as it can capture every single point around the cameras. This feature is potential to eliminate the narrow FoV problem. While the 360 camera can deliver full environment information, displaying the 360 video in the base station can be a problematic. Displaying as a flat image require mental adjustment for the operator to understand the scene as the display will not be natural [10]. To deal with this, the proposed method utilized virtual reality (VR) headset to display the 360 video streaming. With this approach, the operator can see the scene in more natural way and it will also improve the immersive experience as the operator feel they are on the location. This approach also can potentially improve the performance of the teleoperation.

2. System design
To enable teleoperation of robots, a framework has to be developed to accommodate some features, such as:

- Provides a connection between robot to base station via local area network. The communication media must be wireless to ensure unconstrained motion;
- The base station should be able to accept robot control command from the operator, using a controller. In this case, the controller was a game pad;
- The control command then should be able to be transmitted to the robot via local area network;
- The robot should be able to receive this control command and performs motion as instructed;
- In the same time, the robot must be able to capture image around the robot and send it as video streaming to the base station via local area network;
- The base station should be able to receive the video stream and display it on the screen or in the VR headset;
- The protocol for control command as well as for video streaming has to be selected carefully to minimize latency.

The developed framework can be seen in Figure 1.
The robot platform used in the experiment was 4 wheeled, and configured as differential wheel mechanism. Each of the 4 wheels has its own motor and those 4 motors were grouped into 2 groups: left motors and right motors. Using this configuration, only 2 motor drivers were needed. The 2 motor drivers were connected to an Arduino Nano which provided PWM signal for each group. The Arduino Nano (served as low-level controller) was connected to the Raspberry Pi as high-level controller through serial UART communication. The communication between Arduino Nano and Raspberry Pi used ROS serial protocol.

The Raspberry Pi as high-level controller used Ubuntu 16.04 as its operating system. The Ubuntu operating system managed the Wi-Fi connection to the WLAN so the robot can connect to the base station. The Raspberry Pi also implemented Kinetic Robotic Operating System (ROS). The ROS provides robot command communication using standard velocity command messages (cmd_vel). The velocity command was received from base station which also implemented ROS. In the same time, the Raspberry Pi captured image from web camera and send it to base station using standard ROS image transport class. The base station subscribed to the compressed image transport to reduce latency. While webcam video streaming was done using ROS image transport, the 360 streaming was done by a custom plugin for the Ricoh Theta V 360 camera. The plugin provided WebRTC web server that then accessed from Oculus Quest VR browser.

3. Experiment methods
We evaluated these designs with 28 trials by 7 participants. Each trial done by teleoperating robots on a predefined mission. A ceiling mounted camera tracked an AprilTag marker placed on the robot to precisely identify the 2D pose of the robot within the arena. The ceiling mounted camera capable to capture the whole arena and to ensure the accuracy, the ceiling mounted camera was previously calibrated using a checkerboard. The trajectories of each trial were recorded in rosbag and then analysed its performance in Matlab using a standard performance metrics as introduced by (Munoz Ceballos, Valencia Velasquez, & Ospina, 2010). While the full performance metrics are:

1) Mission success: number of successful missions
2) Collisions: number of collisions per mission
3) Path length: distance travelled to accomplish the task
4) Obstacle clearance: minimum and mean distance to the obstacles
5) Robustness in narrow spaces: number of narrow passages successfully traversed
6) Smoothness of the trajectory
7) Time taken to accomplish the task

Some metrics are un-relevant, such as mission success (all trials were always successful), “robustness in narrow spaces” (no “narrow spaces”) and “obstacle clearance” (no obstacles in the arena setup). In final, we measured the performance only using 4 metrics: 1. Collision area; 2. Path length (distance travelled to accomplish the task); 3. Smoothness of the trajectory; 4. Time taken. To minimize the effect of learning, each participant was trained to get familiar controlling the robot. The robot and arena can be seen in Figure 2.
Figure 2. The robot arena (left) and a participant performed a trial in teleoperating the robot(right).

The example results of trajectories, plotted in Matlab can be seen in Figure 3.

Figure 3. The obtained trajectories plotted and analysed in Matlab.

4. Results and discussions
The recorded trajectories were analysed in Matlab and the results then plotted as box plot. The box plot provides more insight to the data as it can display the median, 25th – 75th percentiles, maximum-minimum data and outliers (if any). Since it can display median, percentiles and mix-max data, the boxplot can be used to measure accuracy (from the median) and precision (from the dispersion of dataset).

4.1. Collision area
The boxplot of the collision area of teleoperating robot using webcam and 360 camera is presented in Figure 4 (left). While the median of the data both webcam and 360 camera was very close, the boxplot shows the teleoperation using 360 camera was significantly more precise as shown by a lower deviation. It shows the webcam operation was harder to avoid collision and the teleoperation depends significantly by the skill of operator. In contrast, using 360 camera as feedback is more consistent and the median was also lower (0.255 m²) compared to webcam (0.268m²).

4.2. Deviation from ideal path
The trials should follow the ideal path, which is the middle of the way path. However, most of the teleoperation will not perfectly follow the desired path and it induces longer taken path. Again, the boxplot shows the teleoperation achieved more precise and more accurate. The accuracy can be seen from the median of deviation that scored 0.996m on 360 camera teleoperation, compared to 1.06m on webcam teleoperation as shown in Figure 4 (right). The precision is significantly better as the deviation of 360 camera operation was lower.
4.3. Smoothness
The comparison of smoothness both teleoperation modes is presented in Figure 5. While 360 camera feedback achieved significantly smoother trajectories (lower medians), however the consistency can be seen similar with webcam feedback.

4.4. Time
While aforementioned performance metrics shows that 360 camera feedback system is superior than webcam camera feedback, the last performance metric shows an opposite result. The operation time of 360 camera feedback system took significantly longer than webcam at 74.8 seconds (median) and 61.7 seconds (median) respectively. Teleoperation using 360 camera also more inconsistent in time as shown by larger percentiles. This phenomenon took place due to 360 camera streaming had higher latency. In this case, the operator cannot force the robot to move faster as the real robot position and the video streaming can be unsynchronized. The operator then unwillingly to drive the robot slower. Some disturbance such as intermittent connection also forced the operator to wait until the video streaming getting normal which induced larger time deviation.
5. Conclusions
The experiment demonstrated that the developed framework successfully accommodated the teleoperation. The experiments also proved that teleoperating robot using limited FoV camera (webcam) achieved bigger error from ideal path, while teleoperating robot using 360 camera was closer to the ideal path. A robot teleoperated using webcam tend to collide the obstacles, while teleoperated robot using 360 can better avoid the obstacle. Using 360 camera combined with VR headset also achieved a smoother trajectory. However, due to higher latency of 360 video streaming, the operation time of the proposed method was longer than traditional approach. Therefore, the proposed framework demonstrated good for teleoperating robot in mission critical task while the typical approach (using limited FoV camera) is better in time critical task.

Acknowledgments
Authors wishing to acknowledge financial support from the State Polytechnic of Malang

References
[1] Katyal K D, Brown C Y, Hechtman S A, Para M P, McGee T G, Wolfe K C, Murphy R J, Kutzer M D, Tunstel E W, McLoughlin M P and Johannes M S 2014 Approaches to robotic teleoperation in a disaster scenario: From supervised autonomy to direct control IEEE International Conference on Intelligent Robots and Systems pp 1874–1881
[2] Mostafa S A, Ahmad M S and Mustapha A 2019 Adjustable autonomy: a systematic literature review Artif. Intell. Rev. 51(2) pp 149–186
[3] Mahmoud Zadeh S, Powers D and Zadeh R 2019 Augmented Reactive Mission Planning Architecture: Augmented Reactive Mission and Motion Planning Architecture Cognitive Science and Technology pp 95–107
[4] Asami R, Sawai Y, Sato N, Morita Y, Endo T, and Matsuno F 2016 Teleoperation system with virtual 3D diorama for moving operation of a tracked rescue robot International Conference on Advanced Mechatronic Systems, ICAMechS 0 pp 90–95
[5] De León J, Garzón M, Garzón D, Narváez E, Del Cerro J, and Barrientos A 2016 From Video Games Multiple Cameras to Multi-robot Teleoperation in Disaster Scenarios Proceedings - 2016 International Conference on Autonomous Robot Systems and Competitions, ICARSC pp 323–328
[6] Keyes B, Casey R, Yanco H, Maxwell B, and Georgiev Y 2006 Camera Placement and Multi-Camera Fusion for Remote Robot Operation IEEE International Workshop Safety, Security,
Rescue Robot

[7] Seo S H, Rea D J, Wiebe J, and Young J E 2017 Monocle: Interactive detail-in-context using two pan-and-tilt cameras to improve teleoperation effectiveness RO-MAN 2017 - 26th IEEE International Symposium on Robot and Human Interactive Communication 2017 pp 962–967

[8] Nielsen C W, Goodrich M A, and Ricks R W 2007 Ecological interfaces for improving mobile robot teleoperation IEEE Transactions on Robotics 23(5) pp 927–941

[9] Hedayati H, Walker M, and Szafir D 2018 Improving Collocated Robot Teleoperation with Augmented Reality Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction pp 78–86

[10] Walker M E, Hedayati H, and Szafir D 2019 Robot Teleoperation with Augmented Reality Virtual Surrogates ACM/IEEE International Conference on Human-Robot Interaction 2019 pp 202–210