Teleoperation in Remote Controlled (RC) Vehicles

Amirabbas Sherafatian
University of Tartu
Institute of Computer Science
Tartu, Estonia
Email: amirabbas.sherafatian@ut.ee

***Abstract*—Teleoperation is a technology to operate a machine remotely. It has applications in many areas such as autonomous driving, sidewalk vehicles for delivery purposes, or even telerobotics. As teleoperation connects incompatible components, to make a remote machine accessible, many challenges will emerge.**

This paper tries to highlight the most significant challenges. Besides, it will discuss the classifications of teleoperation together with why teleoperation is needed. Because one of the inevitable parts of teleoperation is camera streaming, a solution to video streaming is also provided in this paper.

Keywords: Teleoperation, Autonomous Vehicles, Teleoperation Challenges, Delay Strategies, Video Streaming.

I. INTRODUCTION

THE term teleoperation conveys the meaning of controlling a remote machine, robot, or system from a distance, While the distance can vary from short distances to thousands of kilometers. Teleoperation dates back to the 19th century as Wikipedia [1] says: ”The 19th century saw many inventors working on remotely operated weapons (torpedoes)”. One of the historical and impressive examples is the robot Sojourner[2] landed on Mars on July 4, 1997, and was alive for 95 days (see the Figure 1). This example truly implies the applications of teleoperation in unreachable, dangerous, or contaminated environments.

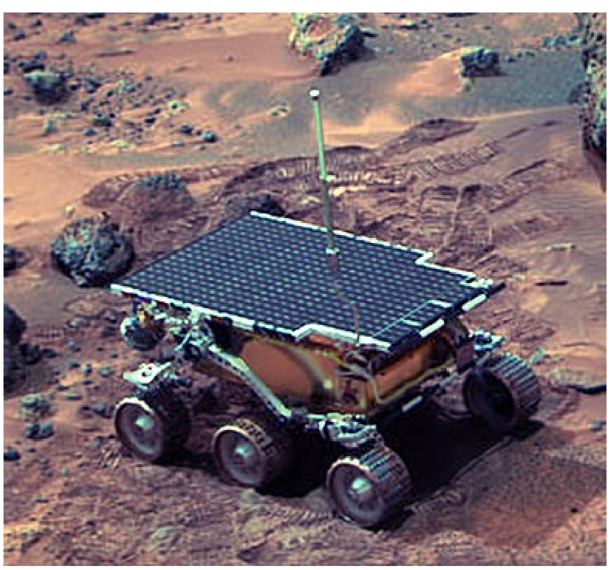

Fig. 1: Sojourner [2]

A. Teleoperation Applications

Indeed, there are many places like deep in the oceans or underground threatening human lives, but they are points of interest for discovery and exploration. Employing robots in such environments, while they are capable of being operated remotely, is an exciting approach. This conveys why teleoperation is signifcant and also points out the applications of teleoperation in industrial machinery and Remotely Operated Vehicles (ROVs).

Teleoperation is also applicable in remote surgery [3] and unmanned aerial vehicles (drones). Remote surgery is another area benefits from teleoperation (see Figure 2) and it has added accuracy and precision to patients’ operations. There are teleoperated arms controlled by surgeons, who are known as operators to send the data to the robot ends. Similarly, drones are also considered teleoperated robots for several purposes ranging from military to delivery services.

Fig. 2: Telesurgery [4]

The use cases of teleoperation are not limited to the areas mentioned. We all have seen and utilized many devices in our daily life that are considered entertainment systems. These devices such as televisions, VCRs, DVD players, etc. are often controlled remotely via remote controls and introduce another category that benefits from teleoperation.

B. Teleoperation Components

Teleoperation consists of two main categories of local components and remote components. Considering the mission, several parts fit into these two main categories. Figure 3 shows a high overview of these components.

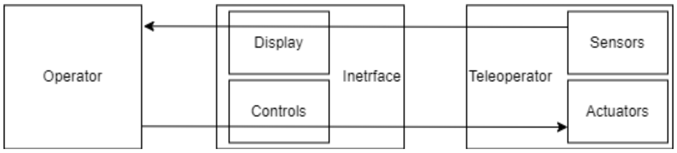

Fig. 3: Teleoperation Components

As Figure 3 represents the components involved, Operator and Interface are components residing locally. Besides, Teleoperator stays in remote sites. The arrows in the figure also demonstrate the communication between the components.

As the Operator is categorized in local components, it represents the person, who sends the command to the machine. The Interface component, at a high level, involves two necessary parts of Display and Controls, where the Display depicts the feedback from the machine. So that the operator will be able to send the right decisions based on that, via Controls.

In the remote end, Sensors and Actuators are inevitable parts of the Teleoperator to respectively transfer the signals to the Interface and convert the decisions received to actual work.

II. DEFINITIONS

a) Teleoperation.: Wikipedia [1] perfectly describes teleoperation: ”Teleoperation (or remote operation) indicates operation of a system or machine at a distance. It is similar in meaning to the phrase ’remote control’ but is usually encountered in research, academia and technology. It is most commonly associated with robotics and mobile robots but can be applied to a whole range of circumstances in which a device or machine is operated by a person from a distance.”

b) Operator.: The person who monitors and sends necessary commands as decisions to the machine – in particular circumstances a human needs to take control of the machine (edge cases) e.g. when the machine needs to cross the street. It is for safety and it is also mentioned as a rule in regulations to have a person taking control of the robots fully in particular situations. This also conveys another reason why we need teleoperation for autonomous vehicles.

c) Actuators.: A machine component to convert control signals into mechanical motion e.g. opening/closing a valve - An actuator requires a control signal and a source of energy [5], [6].

d) Teleoperator.: Teleoperator describes the machine that is remotely operated or the teleoperated machine. Sophisticated teleoperators are also known as telerobots.

e) Autonomous Vehicles.: Autonomous vehicles are self-driving cars and the Gartner glossary defines: ”An autonomous vehicle is one that can drive itself from a starting point to a predetermined destination in ’autopilot’ mode using various in-vehicle technologies and sensors, including adaptive cruise control, active steering (steer by wire), anti-lock braking systems (brake by wire), GPS navigation technology, lasers and radar.” [7].

III. CLASSIFICATIONS OF TELEOPERATION

Teleoperation as a solution to remotely operate vehicles are classified mainly into three categories or levels [8]. Here the classifications of teleoperation are explained.

A. Closed-loop Control

This category is also known as ”Direct Teleoperation”. At this level, there is no autonomy on the remote end and this is the operator who controls the machine from remote distances. The operator receives the feedback via user interfaces and makes the appropriate decisions and sends them to the remote components via the controls available in the user interfaces. In the remote end, the actuators receive the decisions and make the robot operate as it has been decided. Then the feedback on the current situation is captured by the sensors (mostly cameras) and is sent to the user interfaces. This category is known as closed-loop and there is no autonomy in the remote machines. A requirement for this level is to have less or no delay, otherwise, this level is not suitable to be implemented. In case of some delay, there are strategies in conjunction with this level such as ”Move and Wait” and ”Speed Control” that considering the task, may be applied.

B. Coordinated Teleoperation

This level is still similar to the Closed Loop category but there is an additional control in the middle, which adds autonomy to the remote machine to some extent. This additional control is known as a ”remote loop”. The remote loop will help the remote machine with some autonomy when there is some delay. As a result, for specific moments the machine can do some part of the task autonomously. It is important to note that still the majority of the task is done at the operator’s end.

C. Supervisory Control

In Supervisory Control, most of the task is done at the teleoperator end autonomously and this is the difference between this category of teleoperation and the category ”Coordinated Teleoperation”. In this category still, the operator is available and necessary for some moments, but the operator will send high-level commands at some point to the machine. As there exists a higher level of autonomy in this category, the machine is capable to get the majority of the mission done autonomously.

IV. AUTONOMOUS VEHICLES

Teleoperation helps autonomous vehicles to be backed by an operator to get the job done in tough conditions. Autonomous vehicles are mostly equipped with cameras as means to send back the feedback to the operator. As a result, these vehicles can have nobody inside. There are some companies offering remote driving systems as a solution to autonomous vehicles teleoperation e.g. Phantom Auto, Starsky Robotics, and Voysys.

A. Sidewalk Autonomous Vehicles

Delivery services are one of the popular areas employing autonomous vehicles to deliver meals, parcels, packages, and more. These types of vehicles go through the path to the destinations and are capable of interacting with the environment and pedestrians. These vehicles are known as sidewalk autonomous vehicles.

Kiwibot [9] from the company "Kiwi" (see Figure 4) and Postmates from "Serve Robotics" [10] are examples of these kinds of vehicles. They are similar to each other and both are semi-autonomous since they are capable of being teleoperated. Similarly, both use sensors to interact with the environment and a camera to transfer the feedback to the operator. Both need the operator for particular tasks; for instance, the Postmates when needs to cross the street an operator needs to take control of it. Also, for fixing the order they both need assistance.

Fig. 4: Kiwibots

Similarly, Amazon has also introduced Prime Bot [11], which is another example of sidewalk AVs. But the difference between the Prime Bot and the aforementioned examples is that it is fully autonomous, while it is monitored. Amazon has gone further in the area and has had progress introducing a drone as Prime Air to join its delivery fleet (see Figure 5).

Fig. 5: Amazon Prime Air [12]

As the last example, it is worth mentioning the two-legged robot [13] from Agility Robotics [14] that can carry items up to 18KG (see Figure 6). the Ford, as their first customer [15], has employed the two-legged robot and combined it with their autonomous vehicle, so they have introduced an exciting solution as a delivery service. In their solution majority of the journey is traveled by their autonomous vehicle and when it reaches the destination sends a signal to the robot wirelessly. Then the robot jumps off the vehicle to travel the short remaining distance to the recipient's doorstep. The two-legged robot can realize the obstacles on the way to adjust its path. It has also capable of being teleoperated.

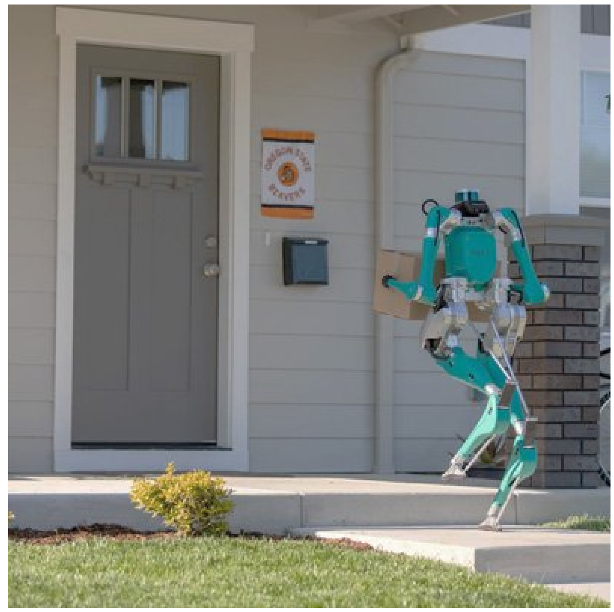

Fig. 6: Two-legged robot from Agility Robotics [14]

V. Challenges

Teleoperation involves many challenges and depending on the mission, more or fewer challenges may emerge. In this section, three main challenges of Connections, Incompatibility of components, and Delay are discussed.

A. Connection

As teleoperation is considered to connect components distributed in different places, The connection is an issue underlying several challenges.

a) Connection Speed.: The connection speed is the first challenge to tackle. When the solution is going to be published, the Internet is a solution to link components together. Consequently, the connection speed will be a focal point. Some may consider utilizing the fast speed connection such as 5G, but there are regions still supporting 4G, or 3G. Indeed, If vehicles need to go far into those areas, some advantageous techniques and architecture need to be thought e.g. compression techniques to compress the request from/to the robots.

Moreover, at some point, vehicles may go into areas with poor connections causing connection loss. For sure, vehicles should be equipped with a mechanism to be prepared for such situations. Depending on the task, applying more levels of autonomy is an option that is considered. As a result, vehicles can do the task autonomously to some extent for the moments that they are disconnected.

b) Bandwidth.: The bandwidth is another challenge when vehicles need to transfer and receive a huge amount of data continuously. This will be more prominent when vehicles need to stream the camera as feedback to the operator. The very first solution may be to reduce the quality of the camera and apply some networking techniques with the help of quality of Service (QoS).

c) Wifi Conflicts: Wifi Conflict is another issue under the connection challenge that we should get ready for it. This usually happens when the project is going to be run in a wireless local area network. In some places, there might be many access points and routers interfering. The solution need to be aware of such situations and may be equipped with some techniques to isolate the connection channel.

B. Incompatibility

Incompatibility between the components connected together is another issue behind the connection. As heterogeneous components are linked together by teleoperation and are wholly introduced as an integrated solution, some challenges appear. As video streaming is a usual mechanism to send feedback to the operator, we take it as an example to explain further.

Indeed, the user interface has a display to show the images received from the teleoperator and they are rendered via an engine. With this in mind, sensors on the teleoperator stream the camera. As a result, the sensors and the engine perform their own tasks separately at different speeds. To have a successful solution, we need to have a mechanism to coordinate and synchronize these two components. In such cases, Buffering and adding some delay are advantageous options that need more investigation to implement based on the mission.

C. Delay

As teleoperation solutions consist of several components, each component adds some amount of delay. For example, the interface adds some delay when the engine wants to render the camera images or the controllers want to receive and send the decisions from the operator to the actuators. Similarly, some delay is added in the teleoperator end when actuators receive the decision to convert them to the actual work. More importantly, the connection is also adding delay to the whole solution.

Considering the user interface and the rendering engine delay, there are companies providing frameworks to cure the delay. For instance, Voysys [16] is a company that has introduced its own rendering engine based on C++ that supports 3D. While it is customizable and it is possible to build our own 3D feature on top of that, it can be integrated into any machine, truck, car, robot, etc [17]. As a consequence, some companies such as Einride [18] have the tendency to integrate their solution into enhanced teleoperation systems - Einride is the company manufacturing self-driving vehicles and is a customer of Voysys for their teleoperation systems.

When there is a delay, then the category of Closed Loop is not an option. While there are ongoing efforts to cope with the delay, increasing the autonomy at the teleoperator end is a workaround to reduce the delay. But there are cases where it is not doable to add autonomy. In these cases, strategies such as ”Move and Wait” and ”Speed Control” will help. In the next section, these strategies are introduced.

VI. STRATEGIES

The two famous Strategies ”Move and Wait” and ”Speed Control” [19] that help with delay are introduced in this section. To explain the strategies, the path shown in Figure 7 is taken into consideration. The path is considered as an example that the teleoperator needs to travel From point A to F. To describe the strategies, the commands ”move forward” (MF), “move backward” (MB), “turn left” (TL), “turn right” (TR), and “stop” (ST) are also defined.

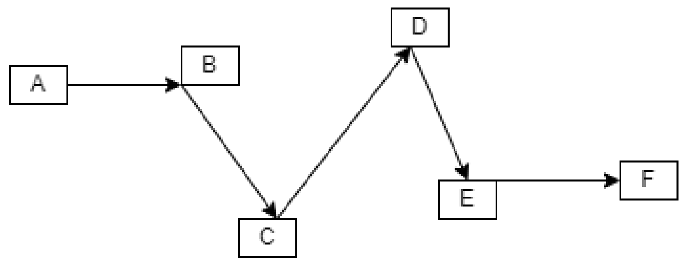

Fig. 7: The path teleoperator needs to travel

A. Move and Wait

In this strategy, to make the teleoperator move toward the destination F, the command MF at the point of A is sent. When it reaches point B, two more commands need to be sent to turn the teleoperator to some degree and move forward. These two commands again need to be sent when the teleoperator is at the points of D and E until it reaches the destination F. Obviously, with parameters sent with the commands, the teleoperator is told how much degree it needs to turn and travel. As a result, we have sent nine commands to drive the teleoperator to the destination. This Strategy is called ”Move and Wait” because the operator needs to wait until the last command is fully executed and after that, the next command can be executed.

B. Speed Control

In this strategy, instead of MF and MB commands, ”No Stop Move Forward” (NSMF) and ”No Stop Move Backward” (NSMB) commands are defined. This means that the teleoperator continues moving forward or backward when receiving these two commands.

Now when the teleoperator is at the point of A, it is enough to remove all the MF commands in the ”Move and Wait” strategies by sending just a single NSMF command. Then at each further point B, C, D, and E, just one command needs to be sent to turn the teleoperator, as it is moving forward continuously. Although, at the point of F, an ST command to stop the teleoperator needs to be sent. As a result, the teleoperator does the task with six operators, and compared to the ”Move and Wait” strategy, the number of commands is reduced.

VII. Video Streaming Solution

As most teleoperators have a camera installed to send feedback to the operator, camera streaming is a big challenge. In this section, we are providing a solution to this challenge while it can be improved further.

From the hardware perspective, we employed a Donkey Car S1 [20], which comes with a Raspberry Pi 4 and a camera. The solution has been written in python and exposes two services on port 8080 so that the client applications are able to make a web socket and HTTP connection and consume the camera. As the code is available on the Github [21], the frameworks ”tornado” and ”picamera” are used.

With the tornado, as a web framework for python, two services are exposed. One of the services is over HTTP and is meant to capture and serve a new image. So that whenever a request is sent to this endpoint, the camera will take an image and serve it through the HTTP protocol. Another service is based on web sockets which are on top of the TCP protocol. For this service, a worker thread captures an image and serves it to the connected clients in a loop. This means that the camera is streamed as chained images taken continuously.

To test the services, two client applications connect to consume the services. One of the client applications is a windows application that is based on Godot (a gaming platform to provide gaming applications in 2D and 3D). This client application is making a web socket connection to the Raspberry Pi and represents the images to the end user. Another client application is the user browser that is sending the request over the HTTP protocol to receive a single image to represent it to the end user. A TP-LINK router (TD-W8960N) to have our clients and server applications in the same network has been set up. The services are serving images with a resolution of 320X240 and the frame rate is set to 30 fps. The images also have a size of 30 KB.

VIII. Result

As the HTTP requests are made to receive just a single image, there is no delay. The server, expectedly, takes a new picture and serves it through the HTTP connection.

For the client application based on the TCP protocol, the delay is around 40 ms. As the log (see Figure 8) shows, the time taken for the images to get rendered is around 40 ms. The delay is acceptable to provide a practical solution, but it can be improved with some techniques such as applying encoders and decoders in the server and the client application.

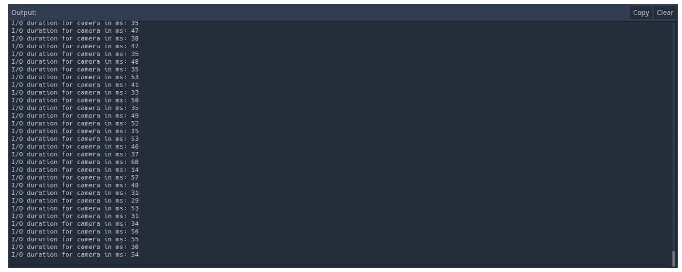

Fig. 8: Time execution for the client application connected through TCP protocol

IX. Concolusion

To conclude, teleoperation is an inevitable part of autonomous vehicles, since in some critical moments it is necessary to have an operator who takes the control of the teleoperator fully. This is also in the regulation to have an operator as a backup. Also, teleoperation is significant and needed, as it makes humans capable to discover new, dangerous, or contaminated areas. Depending on the task, there will be many challenges along with the way and one of the problematic ones is the streaming camera. For this specific challenge, the provided solution helps to start a further investigation. With further study, it is possible to achieve higher resolution, less size, and consequently less delay.

References

[1] Teleoperation. https://en.wikipedia.org/wiki/Teleoperation.

[2] Sojourner (rover). https://en.wikipedia.org/wiki/Sojourner_(rover).

[3] Mohan A, Wara U, Arshad Shaikh M, et al. Telesurgery and robotics: An improved and efficient era, March 2021. Cureus 13(3): e14124. doi:10.7759/cureus.14124.

[4] What is remote surgery/telesurgery? https://www.news-medical.net/health/What-is-Remote-SurgeryTelesurgery.aspx.

[5] What is an actuator. https://www.haroldbeck.com/what-is-an-actuator/.

[6] Actuator. https://en.wikipedia.org/wiki/Actuator.

[7] https://www.gartner.com/en/information-technology/glossary/autonomous-vehicles.

[8] https://www.techopedia.com/definition/15032/teleoperation.

[9] https://www.kiwibot.com/.

[10] https://www.serverobotics.com/.

[11] Sasha Lekach. Amazon's drones still aren't here, but this little robot is now delivering packages. https://mashable.com/article/amazon-delivery-bot-scout, January 2019.

[12] Amazon is still struggling to make drone deliveries work. https://www.theverge.com/2022/4/11/23020549/amazon-struggling-drone-deliveries-prime-air-bezos.

[13] Charlotte Jee. A two-legged delivery robot has gone on sale—and ford is the first customer. https://www.technologyreview.com/2020/01/06/15/a-two-legged-delivery-robot-has-gone-on-saleand-ford-is-the-first-customer, January 2020.

[14] Agility robotics. https://agilityrobotics.com/.

[15] Sasha Lekach. Watch ford's two-legged robot walk packages to your door. https://mashable.com/video/ford-agility-robotics-digit-delivery-robot, May 2019.

[16] https://www.voysys.se/.

[17] Company turns r/c car into a real-life racing simulator with teleoperation. https://www.techeblog.com/teleoperation-rc-car/, August 2019.

[18] https://www.einride.tech/.

[19] Zhong Shengtong, Philippe Le Parc, and Jean Vareille. Internet-based teleoperation: A case study - toward delay approximation and speed limit module. https://www.researchgate.net/publication/44961568_Internet-based_teleoperation_A_case_study_-_toward_delay_approximation_and_speed_limit_module, May 2007.

[20] https://www.donkeycar.com/.

[21] https://github.com/sherafatian-amirabbas/donkey-oculus/blob/interior_car_environment_custom_camera_api/PiCamera-API/camera.py.