2022 IEEE International Conference on
Autonomous Robot Systems and Competitions (ICARSC)
Santa Maria da Feira, Portugal – April 29-30, 2022

2022 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC) | 978-1-6654-8217-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICARSC55462.2022.9784769

A Survey of high-level teleoperation, monitoring and task assignment to Autonomous Mobile Robots

Diogo Correia
DEEC, FEUP
Porto, Portugal
diogo.torres.correia@gmail.com

Manuel F. Silva
ISEP-IPP
INESC TEC
Porto, Portugal
mss@isep.ipp.pt

Antonio Paulo Moreira
DEEC, FEUP
INESC TEC
Porto, Portugal
amoreira@fe.up.pt

***Abstract*—Teleoperation of autonomous mobile robots (AMR) is relevant in logistics operations to automate repetitive tasks that often result in injuries to the operator. This paper presents an overview of the systems involved in the current teleoperation scheme where these AMRs are present as well as some works and advances that have been done in the high-level teleoperation field.**

***Index Terms*—High-level teleoperation, autonomous mobile robots, image transmission**

I. INTRODUCTION

The teleoperation of mobile robots is highly relevant when it comes to performing systematic and repetitive tasks or paths that often cause fatigue or injuries to the operators performing them manually, caused by multiple accelerations and decelerations, as it happens in logistics operations. The number of orders has increased significantly over the years, and it is not always possible to hire a sufficient number of people to deal with this issue. Autonomous Mobile Robots (AMR) are already available but there is a need to supervise them to assure its correct operation and intervene when faced with a given situation or a load position different from the expected one.

Aside from that, in the current teleoperation systems, the task assignment is automatic and does not always provide the performance required due to being a problem with multiple constraints. As a result, an operator must assign tasks to the robots and intervene when they can not make decisions autonomously.

This paper is organized as follows. Section II presents some background on the approaches used in the teleoperation's systems. In section III some works in high-level teleoperation are introduced. Some discussion of these works is presented in Section IV and the conclusions are drawn in Section V.

II. BACKGROUND

High-level teleoperation on a mobile robot requires multiple systems to work together to achieve optimal performance, including the mobile robot, its teleoperation, image transmission, and high level command and task assignment. This section demonstrates some of the approaches used in such systems.

978-1-6654-8217-2/22/$31.00 ©2022 IEEE

A. Mobile Robots

Mobile robots have played a significant role in automating repetitive tasks often present in logistics processes, such as loading/unloading and moving loads from one point to another. These robots are usually used at indoor applications and can move autonomously, making decisions regarding the trajectory they should follow based both on their position and the destination they have to reach. To achieve this autonomy, several systems are involved: perception, localization, navigation, and path planning.

Multiple sensors are used for perception to obtain information about its internal state: encoders, gyroscopes, accelerometers, and the Inertial Measurement Unit (IMU); and its surroundings: artificial vision and laser range finders [1].

Several techniques for localization can be used, some of which are map-dependent and others which are not. In the first case, we find localization by natural (walls, doors) or artificial (beacons) landmarks. Other methods in this category include the Kalman Filter (KF) and the Extended Kalman Filter (EKF) [2]; Markov localization [3]; Particle Filter [4] and Map matching algorithms ([5],[6]). In the second case, solutions to the Simultaneous Localization And Mapping (SLAM) problem are found using approaches such as EKF-SLAM [4], FAST-SLAM ([7], [8]), and odometry that takes into account the information given by the wheels' encoders.

When it comes to navigation, the robot can use predefined paths in the environment, such as magnetic or electrical wires placed on the floor, or it can only use the environment, such as laser and natural navigation [9].

The robot must have access to a map in order to plan a path. This map can be represented in a variety of ways, the most common of which is cells decomposition, as opposed to Visibility graph and Voronoi diagram [1]. For planning, more traditional algorithms can be used such as potential fields [1] and random sampling ([1] [10]), or other approaches like Time Enhanced A* (TEA*) [11] and Dynamic Window Approach (DWA) [3]. The path planning problem is still being researched, and several algorithms have been developed over the years, some taking a similar approach to natural processes (Genetic Algorithms (GA) and Ant Colony Optimization (ACO)) and others taking a different approach (Particle Swarm

Optimization (PSO), Dynamic A* (D*), and Model Predictive Control (MPC)) [12], [13] [14], [15].

B. Teleoperation

The most commonly used teleoperation methods are direct teleoperation and supervisory teleoperation [16]. In direct teleoperation, the vehicle relies on the operator's commands to drive and make decisions, whereas, in supervisory teleoperation, the operator monitors the vehicle and assists the robot in making decisions. To efficiently drive remotely, the operator must have the closest feeling to driving the vehicle in person as possible.

This can be accomplished by providing visual perception via cameras, where the Field of View (FOV) of the cameras has a significant impact on operator performance [17]. This data can be displayed in a simple interface or combined with virtual reality to be displayed on Head-Mounted Displays (HMD). A Light Detection And Ranging (LiDAR) sensor can be used simultaneously to cover any blind spots and provide the distance to objects in the robot's surroundings. Sound and haptic or force feedback are two other methods for assisting the operator's perception of its surroundings that can be used in conjunction with visual perception. Another factor influencing the operator's performance is the time it takes to send information between the operator and the robot [18], as well as the variability of the delays [19]. The length of these delays limits the operator's reaction times, causing him to drive remotely at lower speeds [20]. Most teleoperation strategies have made use of Internet communication to send data [21]. When it comes to transmission protocols, User Data Protocol (UDP) is preferred over Transmission Control Protocol (TCP) because it is faster at the expense of not ensuring that all data is transmitted, which can be surpassed by high-rate transmissions.

C. Image Transmission

Efficient image transmission is critical for providing reliable teleoperation because the operator can see not only the robot's view but also the fleet of robots it is supervising in real-time. As a result, delays should be kept to a minimum or, at the very least, be imperceptible to the operator. The higher the image's quality, the longer transmission time is required, being the latter the aspect that should be preferred as stated in [22]. Some compression algorithms can be used in addition to lowering the image's quality and resolution. Some of the most popular are Motion JPEG (MJPEG), H.264/Advanced Video Coding (AVC), and their suitability is determined by the scenarios in which they are used. The former should be used if there is plenty of bandwidth available, whereas the latter can be used with less bandwidth but requires more processing time [23]. H.265/High-Efficiency Video Coding (HEVC) is an improvement of the latter method that performs better compressions at lower resolutions by increasing the computational power required [24].

The protocol used to transmit the images is relevant in the transmission time. Real-time Transport Protocol (RTP) is a network protocol that can be used with either TCP or with UDP to deliver streaming audio and video media over the internet. Because a video is made up of a continuous stream of images, UDP is still the preferred protocol because the loss of a single image may not have a significant impact on the operator's performance.

According to [23], the use of MJPEG along with UDP provided the best transmission times when comparing different combinations of compression algorithms and protocols.

D. High-level command and task assignment

Automating logistics operations requires a system capable of handling the creation of specific tasks related to moving loads and assigning them to robots. Starting with task creation, this can be achieved through a system that records a series of clicks made by the operator in the interface used to monitor the robots, such as 1) selecting the desired load in the image; 2) choosing the robot; and 3) defining a destination to place the load. So, with the task already created, this system should be able to send this sequence of steps to the assigned robot, along with all pertinent information.

To meet these requirements, a connection can be made between the interface, which can be a Supervisory Control And Data Acquisition (SCADA), and the fleet management system. This way, when the operator performs the required clicks to create an order, this information can be passed to the fleet management system which can then communicate with a master-slave architecture, composed of a master and several Programmable Logic Controller (PLC) slaves corresponding to each robot. The master PLC communicates with the slave PLC to assign the task and the slave reports the status of the task execution to the master.

Some planning algorithms can be used to make better use of resources, removing the operator's responsibility for the robot's allocation. These algorithms can be tailored to achieve a variety of goals, such as minimizing completion time, reducing the likelihood of deadlock occurrence, reducing the total distance traveled, or balancing the overall workload of the robot. Some of the most commonly used algorithms in these types of optimization problems are Tabu Search, Genetic Algorithms (GA), and Ant Colony Optimization (ACO), and an application can be found in [25].

III. High-level teleoperation

Some research has been conducted in the field of high-level teleoperation, which enables the supervision or direct control of a robot. Some works are more concerned with advancements in teleoperation, whereas others present work that is used in logistics operations.

A. Improvements on teleoperation

Some work has been done to improve teleoperation, either using raw images from cameras or virtual reality.

The work developed by X. Shen *et al.* provides immersive teleoperation to the operator using components that are easily available [26]. A car is teleoperated by an operator who has

Fig. 1. Software Architecture [26].

a view inside the vehicle provided by a stereo camera (42° FOV) with pan-tilt capability. Thus, the operator's head movements are translated into pan-tilt movements in the camera to replicate the operator's view if he was inside the vehicle. As depicted in Figure 1 a client (operator)-server (vehicle) architecture is used in which two computers communicate with each other over Institute of Electrical and Electronics Engineers (IEEE) 802.11n, allowing for connections via Wireless Fidelity (Wifi), 3G, 4G, or Wireless Local Area Network (WLAN). The video is transmitted in real-time to the operator using RTP and compressed using the H.264 codec and the GStreamer framework. The operator controls the vehicle using a racing wheel and pedal console game controller, with commands sent to the vehicle using UDP for the messages and services present in Robot Operating System (ROS). Furthermore, the latency present in the various connections used is compared, concluding that WLAN allows for better video quality; however, the fact that 3G and 4G have greater range opens the door for their future use in teleoperation. This solution enables immersive teleoperation and opens the door to future network applications such as 5G, which could significantly improve the results. However, using the compression algorithm MJPEG instead of H.264 would reduce the image transmission time.

A. Hosseini *et al.*, evaluates the performance of an interface that uses mixed reality to improve vehicle teleoperation [27]. Three cameras are used to provide a 210° view of the vehicle's surroundings. The captured images are compressed with the H.264 codec and sent over Long Term Evolution (LTE) using UDP. To supplement the image provided to the operator, information from two LiDARs is used to create a topological map with fixed-size cells, as can be seen in Figure 2. The cells are classified as free, occupied, or unknown in terms of the presence of obstacles, and the detected obstacles have the same height because they are captured at the same level. Driving simulations were carried out using the SILAB software [28], which allowed for the introduction of delays inherent in image transmission and commands of 500 and 100 ms, respectively. In these simulations, the operators demonstrated improved task performance, better perception of their surroundings, and less cognitive effort required to complete them. The use of LiDAR data appears to be interesting in terms of allowing a better perception of the surrounding environment and having additional information such as the distance to specific obstacles and mitigating the effect of some blind spots that may exist. As a result, it can be used to keep the operator from colliding with obstacles during teleoperation.

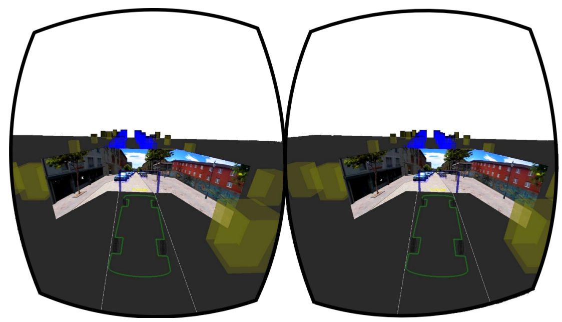

Fig. 2. Perspective view of the mixed reality environment shown within HMD to the human operator [27].

F. Bazzano *et al.*, make a comparison about which is the best camera configuration to use in a robot for telepresence purposes, taking into account its FOV [17]. ROS is used for control, and Javascript is used for the interface, with a library that allows ROS to be used over the web. The operator can control the vehicle in two ways: teleoperation or remote control. In the former, the operator clicks on the image in the web interface, and the coordinates of the point selected by the operator are converted to a point on the real map via ray-tracing. All points on the image are assumed to correspond to positions on the map with axis $z = 0$. As a result, for each pixel (x,y), there is an intersection with the $z = 0$ plane that corresponds to the location to which the robot will move. This movement is accomplished through the use of planning algorithms, and deviation from both fixed and moving obstacles is present. Aside from that, the planned path is shown directly on the video as it moves until it reaches its destination, as showed in Figure 3. The remote control is accomplished by using the keyboard to communicate directly with the robot via the Roslibjs library, allowing the user to change both the orientation and speed of the robot. While the web interface provides the most responsiveness to the operator, it suffers from delays associated with data display and transmission.

Fig. 3. Telepresence interface [17].

The work done by J. Xiao *et al.* uses virtual reality to aid the robot control [29]. The operator interacts with the robot he will control via virtual reality. He has two control modes available to him via a wireless HTC Vive controller: direct and more autonomous. In the former, the controller buttons are used to change the robot's direction and speed. In the latter, the operator points to a virtual position, and the robot calculates the best path to take to reach that point (on the ground) autonomously. ROS is used to control the robot, and the data (commands and images) are encoded in JavaScript Object Notation (JSON) and sent via UDP over a wireless network. The presentation of this system is displayed in Figure 4. To provide this type of interface, intensive processing is required to generate the point clouds, which may result in some delays that may interfere with teleoperation.

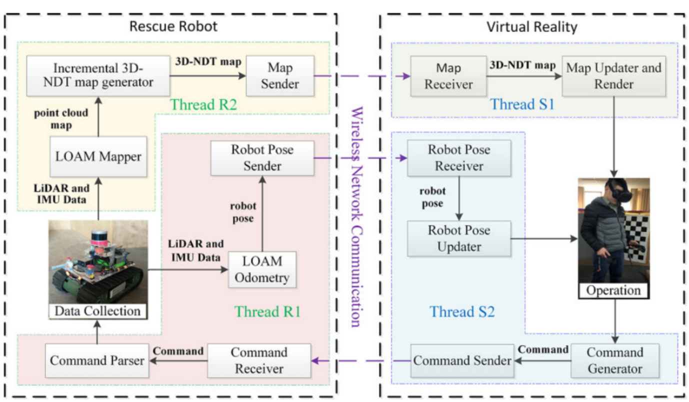

Fig. 4. The proposed Human-Robot Interaction (HRI) method, which is built upon real-time robotic mapping and online Virtual Reality (VR) visualization, where the robot and remote control station communicate with each other by a wireless network. [29].

M.E. Walker *et al.*, implemented a teleoperation's system where the operator controls a drone using an Xbox controller and an Augmented Reality Head-Mounted Display (ARHMD) [30]. Augmented reality shows the operator a "surrogate" as well as an image of the actual drone. They support two modes of navigation: direct and waypoint navigation, as displayed in Figure 5. In the former, the operator directs the "surrogate" to the desired location, and the drone immediately follows, replicating the movement. In the latter, the operator uses the "surrogate" to select the waypoints where the drone should move and then commands the drone to do so. The application is built with the Unity game engine, and commands are communicated via UDP over a wireless connection. Users have improved their path completion time and reduced their cognitive effort as a result of this interface. Although the interface was developed for an aerial robot, it could also be used for mobile robots to provide insight into the robot's future position. If there are significant delays, it could even represent a way to mitigate the impact of delays because the vehicle's current and future positions are given in real-time. Furthermore, if the operator can see the future robot's trajectory, he will be able to predict what will happen to the robot and react accordingly.

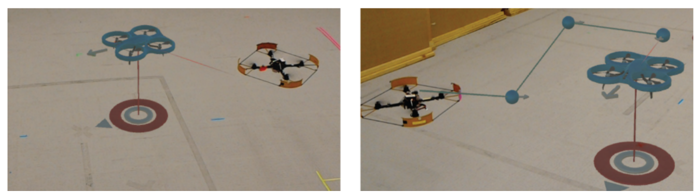

Fig. 5. Two Augmented Reality teleoperation interfaces designed in this work: Left - Realtime Virtual Surrogate (RVS), Right - Waypoint Virtual Surrogate (WVS) [30].

B. Applications in logistics operations

There have already been some forklift teleoperation implementations in logistics operations.

M.C. Mora *et al.* developed a system that allows a fully autonomous forklift to be controlled in two modes of operation: path tracking and teleoperation [31].

Path tracking is the forklift's autonomous mode of operation in which a path is calculated and followed by the forklift based on its initial and destination positions. Because the forklift has implemented the Artificial Potential Fields algorithm along the path, there may be some deviations from the initial trajectory if an obstacle is detected.

Teleoperation is accomplished through the use of a client-server architecture (Figure 6), in which the operator serves as the client and the computer in each forklift serves as the server. The server was developed with Labview and communicates with the forklift's PLC via RS232, sending information to the client such as its orientation and position. If this communication fails, the forklift will remain stationary until the situation normalizes. The client was developed in Darkbasic and receives operator commands, such as orientation via a steering wheel with force feedback and speeds representing the accelerating and braking function. The communication between these applications takes place over WLAN using TCP/IP, resulting in a robust but not very efficient control due to the inherent delay in command transmission and

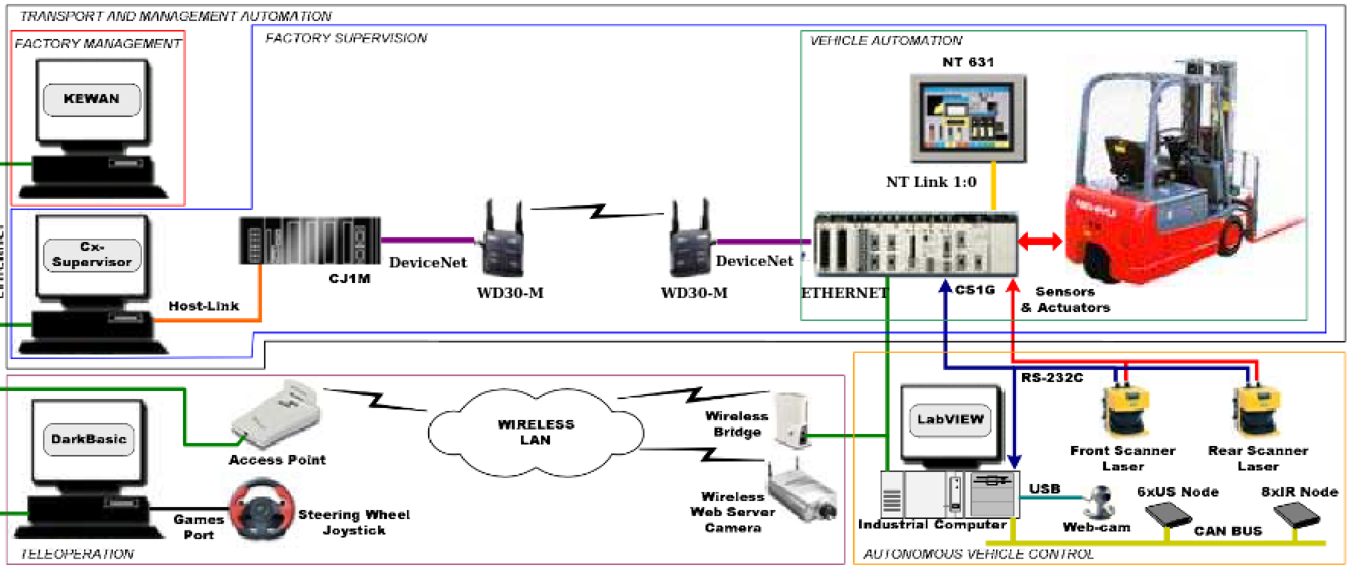

Fig. 6. Hardware architecture (adapted from [31]).

the possibility of retransmissions. A camera with a TCP/IP server is installed in each forklift, which transmits images to the operator at a fixed rate. This paper also presents a high-level system, which involves communication between an Enterprise Resource Planning (ERP) system and a SCADA interface for order assignment and control. Because TCP is used for both control commands and image transmission, the transmission time is longer, whereas UDP would result in faster transmissions.

An hybrid (deliberative and reactive) Autonomous Robot Architecture (AuRA) control architecture is implemented using the MissionLab framework in [32]. Some agents have already been implemented in this framework to control the forklift and provide relevant information to the operator, such as an interface to show the current position, map, and obstacles detected; an order editor; an observer of the robot's state and sensors; robot movement control using a joystick; and communication between all agents. The operator may either take control of the forklift while it performs its tasks autonomously or be requested to intervene if a sensor fails. Nonetheless, the obstacle avoidance agent monitors the teleoperation to prevent the operator from colliding.

H.S. Ahn *et al.*, present a work in which a forklift is controlled by two computers: one in the forklift and the other in the control station [33]. The Player platform employs a client (operator) - server (forklift) architecture, with data transmitted via TCP as depicted in Figure 7. If the latency inherent in the transmission exceeds a certain threshold, the stacker is immobilized until the situation returns to normal. Two threads are used in this architecture: one updates the information sent from the server (direction, speeds) and the other updates information provided by the operator, such as controller handle movement or button presses. A PlayStation 3 controller is used to control the direction, speed, and position

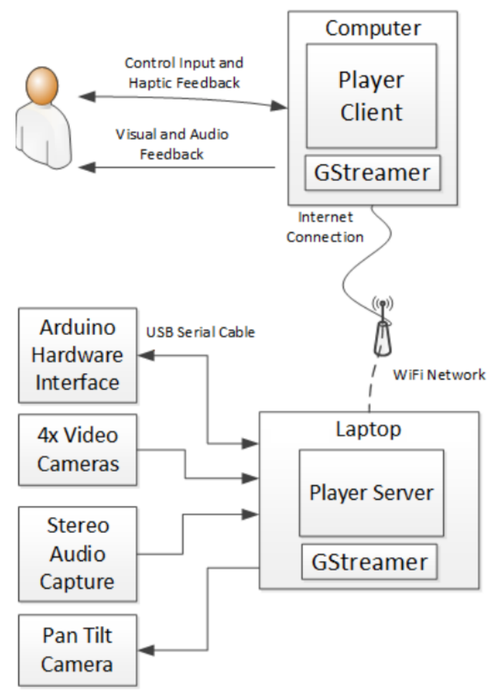

Fig. 7. System architecture of forklift teleoperation system, which consists of a laptop, an Arduino microcontroller, four cameras, a PS3 controller [33].

of the forks, and its vibration (haptic feedback) allows the user to feel the position of the wheel. The GStreamer framework sends stereo audio and images captured by four cameras mounted on the forklift, providing a better perception of the surrounding environment. Each camera was fitted with a 180º fisheye lens to provide a 360º view of their surroundings. One of these cameras has been modified to include pan and tilt functionality. As a result, the operator has the option of observing in one of two directions: downward, to see the path and nearby obstacles, or forward to see a greater distance. These images were transmitted using the GStreamer framework, where they were compressed with the H.264 codec and sent over TCP. The protocol was chosen because it appeared that there were no significant delays detected when compared to the UDP protocol. Furthermore, it is claimed that when the latter protocol was used, there were packet losses and deformations in the video sent to the operator. The data transmission delay between the Player platform and a forklift at a distance of 10 km was measured, yielding an average round-trip time of 110 ms.

C. Surveys on teleoperation

Some works have already investigated the field of teleoperation, aggregating numerous works and the methods used to teleoperate various types of robots.

S. Opiyo *et al.* discuss several works done in the field of teleoperation, including the type of the interface used and the method of connection between operator and robot [21].

In terms of interface, the HMD provides the best sense of presence, but problems such as motion sickness can occur due to poor quality devices (the equipment is relatively expensive) or significant delays between head movement and visualized image.

When it comes to wireless technology (Wifi, mobile network, and Bluetooth), most works adopt Wifi because, while it has a shorter range than mobile network, it is easier to use and consumes less energy. However, with the introduction of the 5G network, the mobile network may become more popular because it not only has a longer range but also a faster data transmission speed, which is critical for teleoperation.

In teleoperation, it highlights that the operator's performance is affected not only from the information received from the environment but also by the reliability of the communication channel. Apart from that, the authors suggest that more research could be done on using Big Data and Machine Learning to help the robot learn its environment. In this way, the robot could provide the operator with decision-making suggestions, reducing cognitive load significantly because it does not need to constantly evaluate the situation.

M.D. Moniruzzaman *et al.* present a comprehensive aggregation of the various methods used in teleoperation and enhancement techniques [16]. As it is stated, the most common teleoperation mode is supervisory control followed by direct control. The former is still evolving and consists of the robot's ability to autonomously plan paths and avoid obstacles. The latter, also known as manual remote control, is the most basic method of teleoperation and is highly dependent on transmission delays. Finally it resumes the enhancements that can be made to teleoperation in 5 main categories, presented in Figure 8, thus being the operator perception, interface, control system, latency and a multimodal teleoperation system. Some of these solutions were used in the works presented above as, for example, in [27] both map merging and mixed reality were used to improve the operator perception and interface.

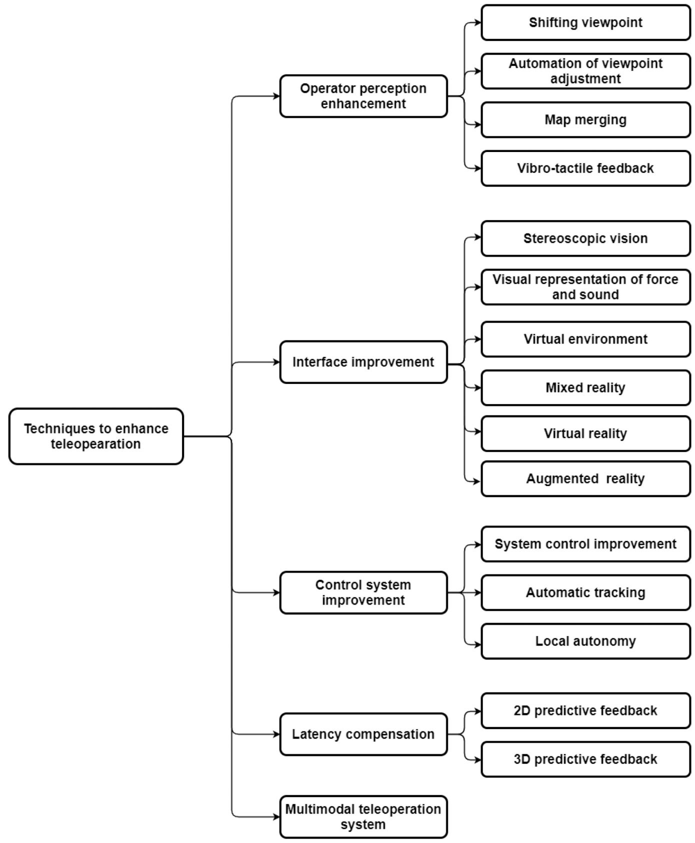

Fig. 8. Classification of teleoperation enhancement techniques [16].

IV. Discussion

As it is seen in Table I, HMDs are used to provide the operator with greater immersion when performing teleoperation. These interfaces, however, require some image processing, which results in longer delays. Furthermore, if there are delays between the images seen by the operator and the movements made with his head, it can create a sense of unease, which does not occur with a flat screen interface. The images sent can be compressed using H.264 to use less bandwidth, as some of the works do, or MJPEG, which requires more bandwidth but takes less time to process. In terms of transmission algorithms, using UDP results in faster transmissions than using TCP, and the mode of communication is mostly via WiFi. However, with the introduction of 5G, this may no longer be the case, as 5G allows for a higher range and faster transmission speeds.

In industry, delays are relevant when performing teleoperation. As a result, a simple interface could be used to reduce processing delays. If this is not an issue, then using HMDs might improve teleoperation performance as long as the equipment has enough quality to match the transmitted images to the operator's movements. To achieve the shortest possible delay MJPEG compression method in conjunction with the UDP protocol can be used, either over WiFi or, if possible, over 5G.

TABLE I
OVERVIEW OF THE TECHNOLOGIES USED IN THE PREVIOUS WORKS

Work	Interface	Communication	Compression algorithm	Transmission protocol
[26]	Head mounted display	WiFi	H.264	RTP, UDP
[27]	Head mounted display	LTE	H.264	UDP
[31]	Flat screen	WLAN	Information not available (N.A.)	TCP
[32]	Flat screen	N.A.	N.A.	N.A.
[17]	Flat screen	N.A.	N.A.	N.A.
[33]	Flat screen	WiFi	N.A.	TCP
[29]	Head mounted display	WiFi	N.A.	UDP
[30]	Head mounted display	WiFi	N.A.	UDP

V. CONCLUSION

The problem of automating logistics operations through AMR teleoperation fits in with the current reality, as companies are increasing the demand for these services. This paper provides an overview of the methods used in the AMR teleoperation systems, specifically AMR autonomy, teleoperation methods and interfaces, approaches to image transmission, and some common algorithms used in task assignment. The high-level operation differs from this type of operation in that the operator is in charge of assigning robots to tasks to be completed. This paper includes a survey of some approaches taken in the field of high-level teleoperation and advances that have been made. In the future, communications over 5G networks may be more frequent due to the superior range and transmission velocity. Aside from that, machine learning appears to be an improvement to implement because it would assist the user by recommending which path the robot should take or which task should be assigned to which robot. When it comes to interfaces, the ones based on virtual and augmented reality may become more common, as they provide a greater immersion in the teleoperation performed and thus higher performance.

ACKNOWLEDGMENT

This work has been supported by the European Regional Development Fund (FEDER) through a grant of the Operational Programme for Competitivity and Internationalization of Portugal 2020 Partnership Agreement (PRODUTECH4S&C, POCI-01-0247-FEDER-046102).

REFERENCES

[1] S. S. Ge and F. L. Lewis, *Autonomous Mobile Robots: Sensing, control, decision-making, and applications*. CRC / Taylor & Francis, 2006.

[2] Y. Kim and H. Bang, "Introduction to kalman filter and its applications," in *Introduction and Implementations of the Kalman Filter*, F. Govaers, Ed., Rijeka: IntechOpen, 2019, ch. 2. DOI: 10.5772/intechopen.80600.

[3] R. Y. Siegwart and I. R. Nourbakhsh, *Introduction to autonomous mobile robots*. MIT, 2004.

[4] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT Press, 2005.

[5] H. Sobreira, C. Costa, I. Sousa, *et al.*, "Map-matching algorithms for robot self-localization: A comparison between perfect match, iterative closest point and normal distributions transform," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 93, no. 3-4, pp. 533–546, 2019. DOI: 10.1007/s10846-017-0765-5.

[6] M. Lauer, S. Lange, and M. Riedmiller, "Calculating the perfect match: An efficient and accurate approach for robot self-localization," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 4020 LNAI, pp. 142–153, 2006. DOI: 10.1007/11780519_13.

[7] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "Fastslam: A factored solution to the simultaneous localization and mapping problem," in *In Proceedings of the AAAI National Conference on Artificial Intelligence*, AAAI, 2002, pp. 593–598.

[8] M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit, *et al.*, "Fastslam 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *IJCAI*, vol. 3, 2003, pp. 1151–1156.

[9] *Agv navigation: Methods, comparison, pros and cons - illustrated guide*. [Online]. Available: https://www.agvnetwork.com/types-of-navigation-systems-automated-guided-vehicles.

[10] H. M. Choset, K. Lynch, S. Hutchinson, *et al.*, *Principles of Robot Motion: Theory, algorithms, and implementation*. MIT Press, 2005.

[11] J. Santos, P. Rebelo, L. Rocha, P. Costa, and G. Veiga, "A* based routing and scheduling modules for multiple agvs in an industrial scenario," *Robotics*, vol. 10, no. 2, 2021. DOI: 10.3390/robotics10020072.

[12] K. Karur, N. Sharma, C. Dharmatti, and J. E. Siegel, "A survey of path planning algorithms for mobile robots," *Vehicles*, vol. 3, no. 3, pp. 448–468, 2021, ISSN: 2624-8921. DOI: 10.3390/vehicles3030027.

[13] H.-y. Zhang, W.-m. Lin, and A.-x. Chen, "Path planning for the mobile robot: A review," *Symmetry*, vol. 10, no. 10, p. 450, 2018, ISSN: 2073-8994. DOI: 10.3390/sym10100450.

[14] M. Hoy, A. S. Matveev, and A. V. Savkin, "Algorithms for collision-free navigation of mobile robots in

complex cluttered environments: A survey," *Robotica*, vol. 33, no. 3, pp. 463–497, 2015. DOI: 10.1017/S0263574714000289.

[15] J. Sánchez-Ibáñez, C. Pérez-Del-pulgar, and A. García-Cerezo, "Path planning for autonomous mobile robots: A review," *Sensors*, vol. 21, no. 23, 2021. DOI: 10.3390/s21237898.

[16] M. Moniruzzaman, A. Rassau, D. Chai, and S. Islam, "Teleoperation methods and enhancement techniques for mobile robots: A comprehensive survey," *Robotics and Autonomous Systems*, vol. 150, 2022. DOI: 10.1016/j.robot.2021.103973.

[17] F. Bazzano, F. Lamberti, A. Sanna, and M. Gaspardone, "The impact of field of view on robotic telepresence navigation tasks," *Communications in Computer and Information Science*, vol. 983, pp. 66–81, 2019. DOI: 10.1007/978-3-030-12209-6_4.

[18] I. MacKenzie and C. Ware, "Lag as a determinant of human performance in interactive systems," in *Conference on Human Factors in Computing Systems - Proceedings*, 1993, pp. 488–493. DOI: 10.1145/169059.169431.

[19] J. P. Luck, P. L. McDermott, L. Allender, and D. C. Russell, "An investigation of real world control of robotic assets under communication latency," in *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, 2006, pp. 202–209. DOI: 10.1145/1121241.1121277.

[20] J. Stückler, M. Schwarz, M. Schadler, A. Topalidou-Kyniazopoulou, and S. Behnke, "NimbRo Explorer: Semiautonomous Exploration and Mobile Manipulation in Rough Terrain," *Journal of Field Robotics*, vol. 33, no. 4, pp. 411–430, 2016, ISSN: 15564967. DOI: 10.1002/rob.21592.

[21] S. Opiyo, J. Zhou, E. Mwangi, W. Kai, and I. Sunusi, "A review on teleoperation of mobile ground robots: Architecture and situation awareness," *International Journal of Control, Automation and Systems*, vol. 19, no. 3, pp. 1384–1407, 2021. DOI: 10.1007/s12555-019-0999-z.

[22] M. Riestock, F. Engelhardt, S. Zug, and N. Hochgeschwender, "User study on remotely controlled uavs with focus on interfaces and data link quality," vol. 2017-September, 2017, pp. 3394–3400. DOI: 10.1109/IROS.2017.8206179.

[23] A. Kaknjo, M. Rao, E. Omerdic, L. Robinson, D. Toal, and T. Newe, "Real-time video latency measurement between a robot and its remote control station: Causes and mitigation," *Wireless Communications and Mobile Computing*, vol. 2018, 2018. DOI: 10.1155/2018/8638019.

[24] S. Ramil, R. Lavrenov, T. Tsoy, M. Svinin, and E. Magid, "Real-time video server implementation for a mobile robot," vol. 2018-September, 2019, pp. 180–185. DOI: 10.1109/DeSE.2018.00042.

[25] P. Udhayakumar and S. Kumanan, "Task scheduling of agv in fms using non-traditional optimization techniques," *International Journal of Simulation Modelling*, vol. 9, no. 1, pp. 28–39, 2010. DOI: 10.2507/IJSIMM09(1)3.139.

[26] X. Shen, Z. Chong, S. Pendleton, *et al.*, "Teleoperation of on-road vehicles via immersive telepresence using off-the-shelf components," *Advances in Intelligent Systems and Computing*, vol. 302, pp. 1419–1433, 2016. DOI: 10.1007/978-3-319-08338-4_102.

[27] A. Hosseini and M. Lienkamp, "Enhancing telepresence during the teleoperation of road vehicles using hmd-based mixed reality," in *2016 IEEE Intelligent vehicles symposium (IV)*, IEEE, 2016, pp. 1366–1373. DOI: 10.1109/IVS.2016.7535568.

[28] *Driving simulation and silab*. [Online]. Available: https://wivw.de/en/silab.

[29] J. Xiao, P. Wang, H. Lu, and H. Zhang, "A three-dimensional mapping and virtual reality-based human–robot interaction for collaborative space exploration," *International Journal of Advanced Robotic Systems*, vol. 17, no. 3, 2020. DOI: 10.1177/1729881420925293.

[30] M. E. Walker, H. Hedayati, and D. Szafir, "Robot teleoperation with augmented reality virtual surrogates," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, IEEE, 2019, pp. 202–210. DOI: 10.1109/HRI.2019.8673306.

[31] M. C. Mora, V. Suesta, L. Armesto, and J. Tornero, "Factory management and transport automation," in *EFTA 2003. 2003 IEEE Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No. 03TH8696)*, IEEE, vol. 2, 2003, pp. 508–515. DOI: 10.1109/ETFA.2003.1248741.

[32] F. Rodríguez, J. Rodríguez-Aragón, B. Diego, and V. Rodilla, "Multi-agent cooperation for advanced teleoperation of an industrial forklift in real-time environment," *Advances in Intelligent and Soft Computing*, vol. 88, pp. 57–62, 2011. DOI: 10.1007/978-3-642-19875-5_7.

[33] H. S. Ahn, S. McArdle, G. Sumner, and B. A. MacDonald, "Development of user interfaces for an internet-based forklift teleoperation system with telepresence," in *Proceedings of Australasian Conference on Robotics Automation*, 2014.