Teleoperation of Mobile Robots by Generating Augmented Free-viewpoint Images

Fumio Okura[1,2], Yuko Ueda[1], Tomokazu Sato[1], and Naokazu Yokoya[1]

***Abstract*— This paper proposes a teleoperation interface by which an operator can control a robot from freely configured viewpoints using realistic images of the physical world. The viewpoints generated by the proposed interface provide human operators with intuitive control using a head-mounted display and head tracker, and assist them to grasp the environment surrounding the robot. A state-of-the-art free-viewpoint image generation technique is employed to generate the scene presented to the operator. In addition, an augmented reality technique is used to superimpose a 3D model of the robot onto the generated scenes. Through evaluations under virtual and physical environments, we confirmed that the proposed interface improves the accuracy of teleoperation.**

I. Introduction

This paper describes a teleoperation interface for a mobile robot that provides a human operator with a novel way to grasp the environment surrounding a robot operating at a remote site. Historically, numerous types of mobile robots have been developed and employed for various situations to operate on behalf of humans [1]. The importance of the teleoperation interface has increased significantly, particularly for unknown and/or extreme environments (e.g., disaster areas) with narrow pathways and unknown obstacles. Although there are research fields devoted to the automatic control of mobile robots [2], [3], most practical robots are still operated by human operators using video images captured by cameras mounted on the robot. These include PackBot [4], which was deployed for the surveillance of the Fukushima Daiichi nuclear power plant in Japan after the earthquake in 2011. A human operator should have sufficient control of a robot to prevent it from colliding with its surroundings, while safely and effectively fulfilling its assigned tasks. To achieve successful operations, it is important to determine the best way to represent the field of view surrounding the robot to its human operators, because vision is the most important sense used by humans to grasp the environment surrounding teleoperation tasks. Therefore, there have been numerous studies on image presentation approaches for the teleoperation interfaces of mobile robots [5]–[16].

The existing remote control interfaces for mobile robots are classified into two categories in terms of the image presentation approach used for human operators:

- Interfaces providing a first-person view of the robot.
- Interfaces providing a third-person (bird's-eye) view of the robot.

The two categories of related interfaces and the proposed interface are discussed in the following sections.

This research was supported by JSPS KAKENHI 24700208, 23240024, Grant-in-Aid for JSPS Fellows 25-7448, and "Ambient Intelligence" project granted from MEXT.

[1]Graduate School of Information Science, Nara Institute of Science and Technology (NAIST), Nara, Japan. {fumio-o, tomoka-s, yokoya}@is.naist.jp

[2]JSPS Research Fellow

II. Related interfaces

A. Interfaces providing first-person view

The most common image presentation approach for mobile robot teleoperation is based on using a first-person view [5], which is the scene directly captured by robot-mounted cameras. Most studies and robotic products have employed monocular cameras to provide a first-person view for surveillance in such environments as minefields [6] and sewers [7]. Omnidirectional cameras [17] are often used for a first-person view interface that enables operators to configure their view direction. Although employing omnidirectional cameras reduces the delay when changing the view direction, a couple of problems remain in relation to the operator's understanding of the robot's surroundings:

(a) There are missing areas in the scene as a result of occlusions by the robot itself.
(b) It is difficult to grasp distances from surrounding obstacles.

B. Interfaces providing third-person view

Some interfaces that provide third-person views, which are scenes from above or (diagonally) behind a robot such as a bird's-eye view, are expected to overcome the problems listed above. The mobile robots employed for the surveillance of the Fukushima Daiichi nuclear power plant in Japan were operated through a third-person view interface using a pair of identical robots: one moved forward for surveillance, whereas the other captured images of the first robot from behind [8]. A study by Shiroma et al. [9] provided images of a robot from above by physically mounting a camera on a long arm. Their investigation demonstrated that the third-person view is better than the first-person view from the perspective of speed and safety.

Because it is difficult to capture a third-person view from physically mounted cameras in most situations, image processing and/or multi-sensor integration approaches, which combine information captured from the robot, are often employed to generate third-person views. Time Follower's Vision [10] provides a viewpoint from behind by displaying the images captured several seconds ago when the robot is moving forward. To generate more customized viewpoints, the 3D shapes of objects in the surrounding environment acquired from depth sensors mounted on a robot are often

used with the images captured by a camera. The interfaces proposed by Saitoh et al. [11], Nielsen et al. [12], and Ferland et al. [13] provide operators with both the first-person view and 3D models reconstructed based on SLAM approaches [18]. Kelly et al. [14] realized a photorealistic third-person view interface by appropriately mapping images to 3D shapes in an outdoor environment.

In the ordinary third-person view interfaces described above, the operator's viewpoint is fixed or selectable from a few viewpoints that are configured beforehand. Although these types of third-person view interfaces improve the speed and safety of teleoperation, the problems (a) and (b) mentioned above are still not resolved, particularly in a complex environment such as one containing narrow pathways and obstacles because of the limited viewpoint selection. From this perspective, one of the ultimate forms of third-person view interfaces is a Virtual Environment Vehicle Interface (VEVI) [15], [16], which provides a freely configurable view, i.e., the viewpoint and direction can be freely changed by the operator, using a head-mounted display (HMD) and a head tracker. Although an interface with a freely configurable view is expected to provide intuitive and safe operations, the existing VEVIs [15], [16] have been developed as virtual reality interfaces without using real-world textures.

Unlike conventional VEVIs, this study realizes an intuitive and freely configurable third-person view interface using an HMD and head tracker to provide photorealistic textures of complex, real-world environments. Furthermore, we report the effectiveness of an actually developed free-viewpoint operation interface for a real environment through some evaluations. The approaches used to realize freely configurable views with the textures of a real environment are described in the following.

III. Free-viewpoint image generation

In the fields of computer graphics and computer vision, techniques for generating freely configurable views from multiple images are referred to as free-viewpoint image generation (sometime these are also referred to as arbitrary- or novel-viewpoint image generation). One of the free-viewpoint image generation approaches is known as model-based rendering (MBR). This approach is the traditional computer graphics/vision pipeline that reconstructs the 3D shapes of real environments first, and then maps images of the environment over them as textures. At present, 3D shapes can be acquired in real-time from a small desktop environment [19] to a large outdoor environment [20]. In this approach, the quality of the free-viewpoint images generated by MBR is directly affected by the accuracy of the 3D shapes, i.e., unnatural distortions or missing areas in the views are easily exposed. On the other hand, image-based rendering (IBR) generates free-viewpoint images without using explicit 3D shapes. There have been numerous studies on IBR techniques such as view morphing [21] and light-field rendering [22], [23]. Although IBR reduces the missing areas in the resultant images, this approach requires images captured at a large number of places and directions. Otherwise, large distortions may appear [24]. In recent years, the main direction for the research on free-viewpoint image generation is to use hybrid approaches that combine MBR and IBR [25] with the goal of resolving their respective problems. The state-of-the-art method of hybrid rendering [26] appropriately transforms both the 3D shapes and textures depending on the viewpoint of the images about to be generated.

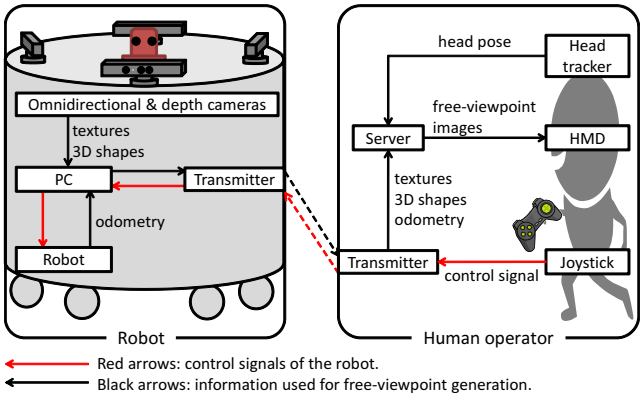


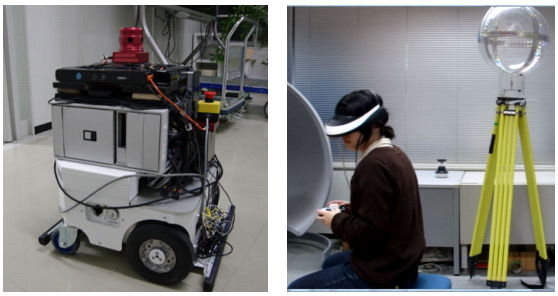
Fig. 1. Data flow diagram of proposed interface.

The proposed interface employs a hybrid rendering method similar to the method in [26]. Because the method in [26] does not achieve real-time processing, we simplify and improve it to realize real-time image generation. In addition, a 3D model of the mobile robot is superimposed on the free-viewpoint images using an augmented reality (AR) technique [27], which is referred to as augmented free-viewpoint image generation in this paper. In the following sections, details of the interface are described with a specific example of a prototype system using an omnidirectional camera and four depth cameras. We also discuss the effectiveness of the free-viewpoint interface through evaluations under virtual and physical environments.

IV. Augmented free-viewpoint interface

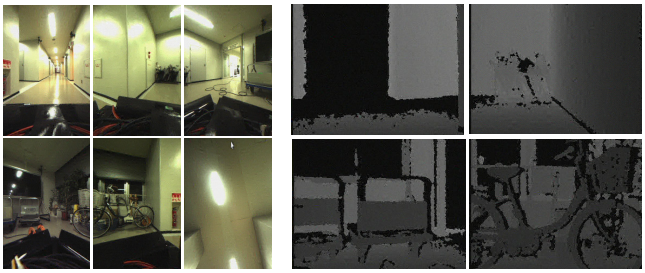
A. Overview

Fig. 1 provides an overview of the proposed free-viewpoint interface. The robot is equipped with a camera and depth cameras for acquiring environmental information such as the textures and 3D shapes of the surroundings. The environmental information and odometry from the wheels of the robot are transmitted to the site of the human operator. A server receives the information from the robot, and generates augmented free-viewpoint images that are displayed on an HMD in real time. The viewpoint of these generated images synchronously changes with the pose (position and direction) of the human operator's head, which is acquired by a head tracker mounted on the HMD. The details of the augmented free-viewpoint image generation method used in the proposed interface are described in Section IV-D. It should be noted that we actually employed a simple wheeled robot operating with control signals from a joystick, including forward/backward movement and rotation. Nevertheless, the



(a) Robot. (b) Human operator.

Fig. 2. Appearance of the prototype system.



(a) Omnidirectional images. (b) Depth images.

Fig. 3. Examples of environmental information.

proposed interface is compatible with other types of mobile robots and control methods.

B. System configurations

In this section, we describe the specifications of the proposed interface using a prototype system shown in Fig 2, which was employed for the experiments in Section VI.

1) Configurations of prototype robot: Fig. 2(a) shows the prototype robot, which has an omnidirectional camera and four depth cameras. The robot (Reference Hardware, Mayekawa Manufacturing Co.) is equipped with four wheels that enable forward/backward movement and rotation. An omnidirectional multi-camera system (Ladybug2, Point Grey Research, Inc.) mounted on the top of the robot captures the textures of the omnidirectional first-person view from the robot. Four depth cameras (Kinect, Microsoft, Inc.) are mounted on four sides to acquire depth images of the surroundings. The relative position and orientation between the omnidirectional and depth cameras are calibrated in advance. Note that the horizontal field-of-view of each depth camera is 57°. This indicates that the four depth cameras cannot cover all of the robot's surroundings at one time. Examples of the environmental information (textures and 3D shapes as depth images) captured by the prototype system are shown in Fig. 3.

In this system, environmental information is newly acquired and transmitted to the server when 1) the moving distance of the robot exceeds a threshold from the latest capturing, 2) the rotation angle of the robot exceeds a threshold, or 3) a certain period of time has elapsed. The acquired images are combined into one large image and transmitted by a wireless HDMI extender with small delays (< 1 [ms]) for the experimental environment.

2) Configuration for human operator: The human operator wears an HMD (HMZ-T1, Sony) that displays augmented free-viewpoint images. This HMD is equipped with a receiver for an electromagnetic sensor (Fastrak, Polhemus, Inc.) that works with a transmitter to measure position and orientation of the operator's head. Augmented free-viewpoint images are generated from environmental information (textures and depth images from the robot), odometry, the head pose of the operator, and the 3D model of the robot in the server. When only one shot of the depth-image-set is used to generate the free-viewpoint images, large missing areas appear due to occlusions, as well as the lack of depth information from the four depth cameras, which do not cover the entire view from the robot. The proposed interface unifies the time-series 3D point clouds to reduce such missing areas. The latest environmental information received from the robot is combined with older information by 3D point cloud alignment using the odometry information as the initial guess for every transmission of environment information. The augmented free-viewpoint images are generated in real-time from 3D point clouds combined by L0-norm-based alignment as well as omnidirectional images. The depth image unification and augmented free-viewpoint image generation processes are described in more detail in the following sections.

C. Unification of multiple depth images

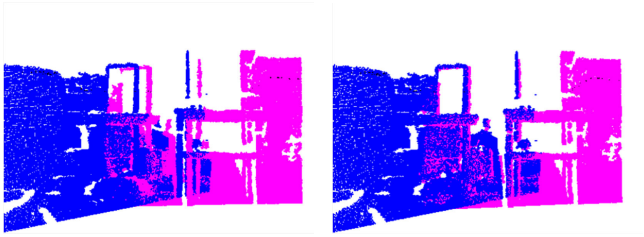
The proposed interface accurately estimates the position and orientation of the robot for each capturing operation by aligning the time-series depth images while minimizing the norms among the multiple 3D point clouds using odometry information as the initial guess.

We employ L0-norm minimization with a two-dimensional exhaustive search algorithm for our prototype system. Using L0-norm is a robust solution for point cloud alignment with large outliers; however, it is difficult to minimize L0-norm using gradient-based minimization algorithms. To achieve a real-time process, the 3D points that exist in a certain interval of height are first projected on a 2D horizontal plane. Then, the minimum value of a cost function based on L0-norm is searched in a 2D solution space by changing the rotation and translation by tiny intervals around the acquired odometry. Although our implementation searches for a minimum in a straightforward manner, it would be possible to use pairs of techniques for efficient searching, such as a SLAM based on L0-norm minimization [28]. Note that because the proposed interface does not specify the alignment methods for point clouds, other alignment techniques can be employed (e.g., modern ICP algorithms [29])

When aligning point cloud $\mathbf{p}$ to another point cloud $\mathbf{q}$, L0-norm $|\mathbf{p}_i, \mathbf{q}_j|_0$ is defined as

$$|\mathbf{p}_i, \mathbf{q}_j|_0 = \begin{cases} 0 & (\exists j, |\mathbf{p}_i - \mathbf{q}_j|_2 \leq \epsilon) \\ 1 & (otherwise) \end{cases}, \quad (1)$$

where ϵ denotes a tiny distance that can be regarded as an identical point. The system minimizes the $E(\mathbf{R}, \mathbf{t})$ defined



(a) Aligned only by odometry. (b) Aligned by using L0-norm.

Fig. 4. Aligned point clouds before and after refinement between those captured from two positions, which are distinguished by their color.

as the sum of L0-norms with changing rotation matrix $\mathbf{R}$ and translation vector $\mathbf{t}$ from $\mathbf{p}$ to $\mathbf{q}$ as

$$E(\mathbf{R}, \mathbf{t}) = \sum_i |\mathbf{R}\mathbf{p}_i + \mathbf{t}, \mathbf{q}_j|_0. \quad (2)$$

The transformation from the robot to the world that is denoted as $\mathbf{R}$ and $\mathbf{t}$ is also used to augmented free-viewpoint image generation.

Fig. 4 shows examples of point clouds with and without the L0-norm-based alignment process. Misalignments between two point clouds are reduced by the L0-norm-based alignment process.

D. Augmented free-viewpoint image generation

Augmented free-viewpoint images are generated from the unified point clouds, operator's viewpoint, and 3D model of the robot in the following three steps:

1) View-dependent geometry generation.
2) View-dependent texture mapping.
3) Superimposition of 3D robot model.

This study employed a free-viewpoint image generation method with view-dependent geometry and texture [26]. Although the method in [26] requires the preliminary reconstruction of 3D shapes using multi-view stereo approaches and does not achieve real-time processing, we use the 3D point clouds acquired using depth cameras and realize real-time processing by eliminating global optimization in the geometry generation and pixel-wise texture selection. In addition, because the free-viewpoint images do not present the appearance of the robot itself, the 3D model of the robot is superimposed using a standard AR technique.

1) View-dependent geometry generation: This process regenerates a depth image of the operator's view using the point clouds to reduce any missing areas and unnatural distortion of the free-viewpoint images. The depth is estimated from the combined point clouds in the following steps.

Step 1: Divide the view plane of the operator into triangular meshes, as shown in Fig. 5. We employ $\mathbf{p}_i$ as the vertices of the meshes.

Step 2: Project the 3D points onto the view plane. Note that some 3D points are far away from the appropriate depth such as those existing over walls.

Step 3: Estimate the depth of each vertex $\mathbf{p}_i$. The depths of the projected points neighboring $\mathbf{p}_i$ are compared, and the depth value d_i of the point $\hat{\mathbf{p}}_i$ that has the smallest depth is employed as the depth of the vertex $\mathbf{p}_i$.

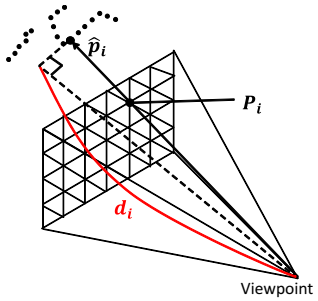


Fig. 5. View-dependent geometry generation.

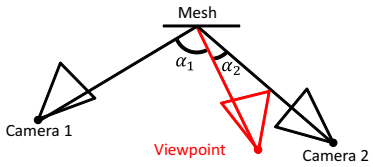
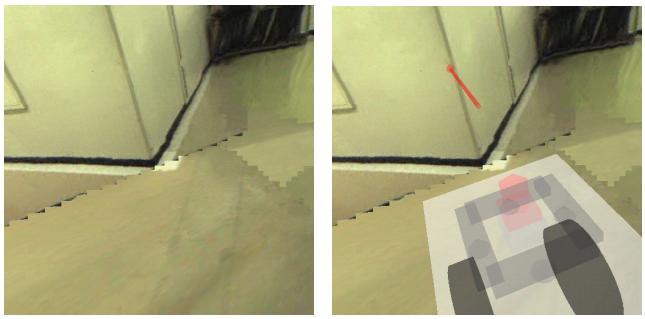


Fig. 6. View-dependent texture selection. Camera 2 is selected as the mesh texture in this case.

It is possible that there are no $\hat{\mathbf{p}}_i$ corresponding to $\mathbf{p}_i$ because of a lack of depth information, which is caused by the occurrence of occlusions in a complex environment, as well as the limited field-of-view of the depth cameras. If there are neighboring vertices whose depth values have been estimated successfully, d_i is determined using linear interpolation of the valid depth values. Otherwise, d_i is set as the largest value that can be measured by the depth cameras (4000 [mm] in our prototype system).

2) View-dependent texture mapping: For each mesh generated in the view-dependent geometry generation, the appropriate texture is selected from the time-series omnidirectional textures captured for the mesh. The generated geometry may include some errors in its 3D shapes. As shown in Fig. 6, we define α as the angle between two vectors from the center of the mesh: one is to the camera capturing the texture of the mesh and the other is to the viewpoint to be generated. The poses of the robot estimated in Section IV-C are used as the pose of the camera. The proposed method selects the texture that has the smallest α because the distortion of the appearance caused by the 3D shape errors is smaller when α is smaller. Finally, the selected texture is projected and mapped onto the mesh. It should be noted that this texture selection strategy is common in some IBR and hybrid rendering approaches [25].

3) Superimposition of 3D robot model: Note that the generated free-viewpoint images do not include the appearance of the robot itself, as shown in Fig 7(a). In our interface, the preliminarily acquired 3D model of the robot is superimposed using the pose information of the robot estimated by the process of aligning the depth images described in Section IV-C. The transmission of the depth images, which requires a sufficiently large bandwidth, may cause large delays. Therefore, the robot may not be superimposed in



(a) Free-viewpoint image. (b) Augmented free-view image.

Fig. 7. Superimposition of 3D robot model.

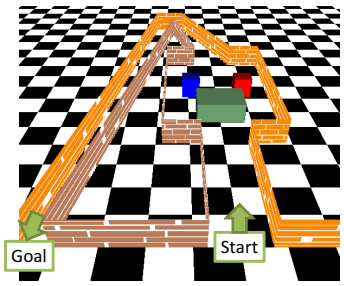


Fig. 8. Virtual environment for experiment.

the appropriate position in a free-viewpoint image when only the pose of the robot estimated in Section IV-C is used. To generate augmented free-viewpoint images while considering such delays, changes in the pose of the robot since capturing the latest depth images are calculated from odometry information and used with the alignment-based pose information. In our prototype system, the 3D model is superimposed transparently to improve the visibility of the scene on the far side of the robot, along with a virtual arrow indicating the traveling direction of the robot, as shown in Fig. 7(b).

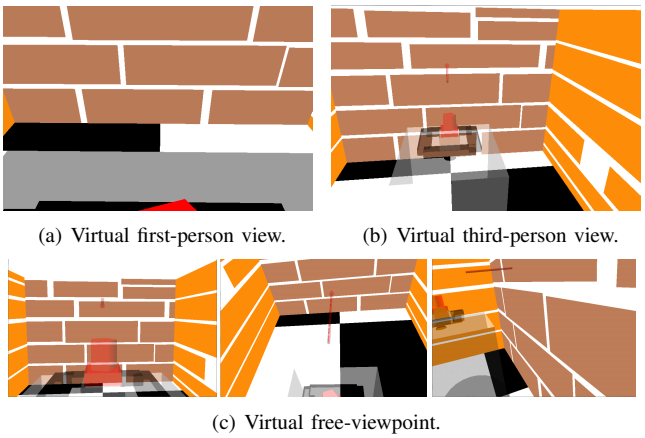
V. Experiments under virtual environment

Under a real environment, there are many factors affecting a human operator's experience, such as those concerned with the quality of the free-viewpoint images generated by the proposed approach. First, we conducted an evaluation to investigate the characteristics of the interface using freely configurable views under an ideal (virtual) environment.

The configuration for the human operator was the same as discussed in Section IV-B.2, and examinees at a remote site operated a virtual robot in a simulation environment using CG models. Ten examinees, who consist of their twenties or thirties, carried out the two tasks below:

Task 1: Run through a path as quickly and safely (so as not to collide with the wall) as possible. The path included narrow passages and some obstacles, as shown in Fig. 8.

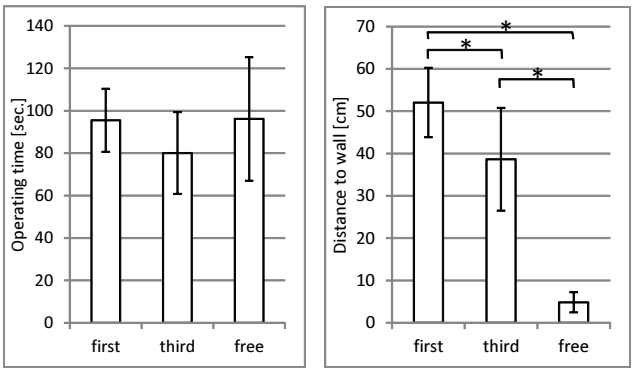
Task 2: Approach the wall as closely as possible, without colliding with it. Such a behavior is sometimes required to accurately operate a robot with arms beside a wall in practical situations.



(a) Virtual first-person view. (b) Virtual third-person view.

(c) Virtual free-viewpoint.

Fig. 9. Examples of view of each interface in virtual environment.

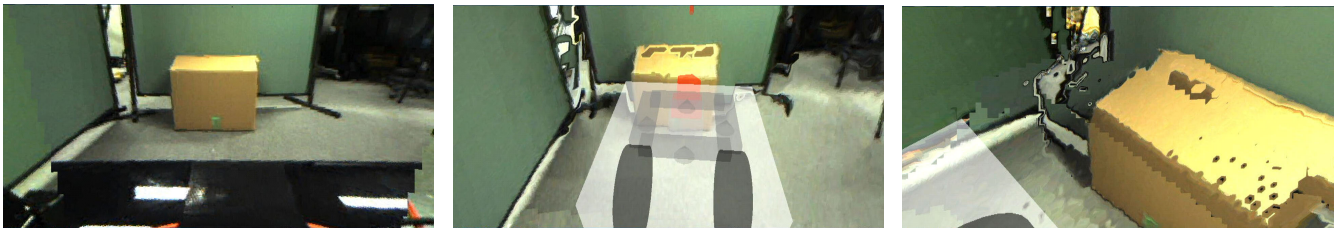


(a) Operating time for first task. (b) Distance to wall in second task.

Fig. 10. Results of experiments in virtual environment: ("*" indicates a significant difference, $p < 0.05$).

For each task, we compared three interfaces: the first-person view, third-person view, and augmented free-viewpoint interface. The third-person view was fixed at a diagonal location behind the robot, approximately $45°$ above the horizontal. The images presented to the operators were generated without the proposed free-viewpoint image generation method, by rendering the virtual environment only from the configured viewpoint using a traditional graphics library, as shown in Fig. 9.

Fig. 10 shows the results of the experiments in the virtual environment, as well as the pairs that had significant difference $p < 0.05$ calculated using a multiple comparison test. We employed the one-way repeated measures analysis of variance (ANOVA) with a Bonferroni-adjusted post hoc paired t-test for the comparison. In the results for the first task (Fig. 10(a)), the operating times to complete the task were not significantly different among the three interfaces. In the second task, the free-viewpoint interface was significantly more accurate than the other interfaces. This implies that the free-viewpoint interface could generate viewpoints that allowed the operators to effectively grasp the distance between the robot and the wall, as shown in the right figure of Fig. 9(c). These experiments indicated that the proposed interface has an advantage related to accurate and safe operation rather than a better operating time.



(a) First-person view. (b) Third-person view. (c) Free-viewpoint.

Fig. 11. Examples of view of each interface in physical environment.

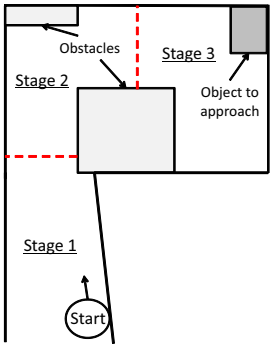
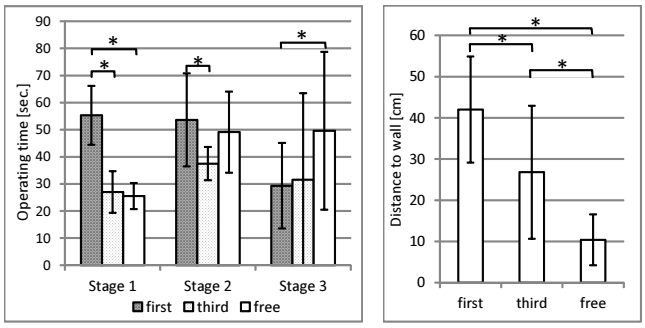


Fig. 12. Map of experimental physical environment.



(a) Operation time of each stage. (b) Distance to wall in Stage 3.

Fig. 13. Time and accuracy of robot teleoperation for each interface ("*" indicates a significant difference, $p < 0.05$).

VI. Experiments under physical environment

We performed experiments in a physical environment using the prototype system described in Section IV-B. Ten examinees, who consist of their twenties or thirties, operated the physical robot using the same three interfaces discussed in the previous section. In this experiment, the first-person view was generated from the latest omnidirectional image, and the examinees could change their view direction freely. The third-person viewpoint was fixed at a diagonal position behind the robot with configurable view direction, whose images were generated using the same technique used in the proposed free-viewpoint interface. Examples of the view of each interface are shown in Fig. 11. The runway for the robot used in the experiment was constructed in three stages, as shown in Fig. 12. The examinees were directed to operate the robot with respect to each stage without collisions with the wall and obstacles. A task was set as follows:

Stage 1: Run through a straight narrow passage as quickly as possible.
Stage 2: Run through a curved passage with obstacles as quickly as possible.
Stage 3: Approach the wall as closely as possible.

We evaluated the operating time for each stage, along with the distance to the wall in Stage 3.

In addition to the objective investigation, we conducted subjective evaluations using the two questions shown below to ascertain the operator's impression of each interface:

Q1: Were the obstacles on the ground easily recognized?
Q2: Was it possible to grasp the distance between the robot and the wall?

The questions were originally formulated in Japanese, and the examinees answered these questions after each stage of the experiment using a scale of one (worst) to five (best).

Fig. 13(a) shows the operating time for each stage and interface. The figure also shows the pairs that had significant difference $p < 0.05$ calculated by the same manner in Section V. In Stage 1, the free-viewpoint interface allowed the operators to complete the task quickly, along with the third-person view interface. In the other stages, significantly longer times were taken to finish the tasks using the proposed interface compared to the third-person view. It is considered that this was because the operators required time to find the most suitable viewpoints when using the free-viewpoint interface. The operation time for Stage 1 was much shorter than the times for the other stages because the examinees mostly fixed their viewpoint at a diagonal position behind the robot when using the proposed interface in Stage 1. On the other hand, the distance to the wall in Stage 3 when using the proposed interface was significantly smaller than with the others (see Fig. 13(b)). These results show the same trends as the experiments under the virtual environment: the proposed interface improved the operation accuracy rather than the time. The occurrence of the same trends in ideal and physical environments indicates that the free-viewpoint image generation process in our prototype system successfully expressed the potential advantage of the proposed interface.

The results of the questionnaires are shown in Fig. 14. The free-viewpoint interface had higher ratings for both Q1 and Q2, which were questions concerning the ability to recognize the surrounding environment. These results imply that the proposed interface reduces the ambiguity in the recognition of the surroundings by the operator.

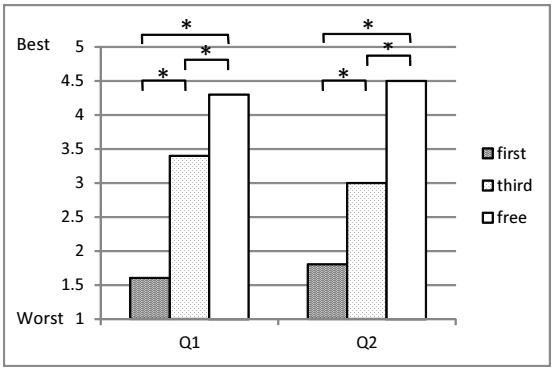


Fig. 14. Result of questionnaires ("*" indicates a significant difference, $p < 0.05$).

VII. Conclusions

This paper has proposed a teleoperation interface with a freely configurable view using photorealistic textures of the physical world for mobile robots. This system allows human operators to change their viewpoints intuitively using an HMD and head tracker. A free-viewpoint image generation method, which is a state-of-the-art technique in the computer graphics/vision research fields, was simplified and improved to achieve real-time processing for the proposed interface. In addition, a 3D model of the robot was superimposed on the free-viewpoint image using AR techniques. This was referred to as augmented free-viewpoint image generation in this paper. We conducted experiments under both virtual and physical environments, and confirmed that the proposed interface has potential advantages in terms of operation accuracy rather than the time required to complete tasks. At the same time, the quality of the generated free-viewpoint images was sufficient to demonstrate the advantage of our prototype system in a physical environment. In future work, we will investigate the effects of delays in the proposed interface under an environment with large delay and improve the prototype system for more practical situations.

References

[1] G. N. DeSouza and A. C. Kak, "Vision for mobile robot navigation: A survey," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 237–267, 2002.

[2] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*. The MIT Press, 2011.

[3] H. Choset, "Coverage for robotics–A survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1, pp. 113–126, 2001.

[4] B. M. Yamauchi, "PackBot: A versatile platform for military robotics," in *Proc. SPIE*, vol. 5422, Unmanned Ground Vehicle Technology VI, 2004, pp. 228–237.

[5] T. Fong and C. Thorpe, "Vehicle teleoperation interfaces," *Autonomous Robots*, vol. 11, no. 1, pp. 9–18, 2001.

[6] D. W. Hainsworth, "Teleoperation user interfaces for mining robotics," *Autonomous Robots*, vol. 11, no. 1, pp. 19–28, 2001.

[7] R. T. Laird, M. H. Bruch, M. B. West, D. A. Ciccimaro, and H. R. Everett, "Issues in vehicle teleoperation for tunnel and sewer reconnaissance," in *Proc. 2000 IEEE Workshop on Vehicle Teleoperations Interfaces*, 2000.

[8] K. Nagatani, S. Kiribayashi, Y. Okada, S. Tadokoro, T. Nishimura, T. Yoshida, E. Koyanagi, and Y. Hada, "Redesign of rescue mobile robot Quince," in *Proc. 2011 IEEE Int'l Sympo. on Safety, Security, and Rescue Robotics (SSRR'11)*, 2011, pp. 13–18.

[9] N. Shiroma, N. Sato, Y. Chiu, and F. Matsuno, "Study on effective camera images for mobile robot teleoperation," in *Proc. 13th IEEE Int'l Workshop on Robot and Human Interactive Communication (ROMAN'04)*, 2004, pp. 107–112.

[10] M. Sugimoto, G. Kagotani, H. Nii, N. Shiroma, F. Matsuno, and M. Inami, "Time Follower's Vision: A teleoperation interface with past images," *IEEE Computer Graphics and Applications*, vol. 25, no. 1, pp. 54–63, 2005.

[11] K. Saitoh, T. Machida, K. Kiyokawa, and H. Takemura, "A 2D-3D integrated interface for mobile robot control using omnidirectional images and 3D geometric models," in *Proc. Fifth IEEE and ACM Int'l Sympo. on Mixed and Augmented Reality (ISMAR'06)*, 2006, pp. 173–176.

[12] C. W. Nielsen, M. A. Goodrich, and R. W. Ricks, "Ecological interfaces for improving mobile robot teleoperation," *IEEE Trans. on Robotics*, vol. 23, no. 5, pp. 927–941, 2007.

[13] F. Ferland, F. Pomerleau, C. T. Le Dinh, and F. Michaud, "Egocentric and exocentric teleoperation interface using real-time, 3D video projection," in *Proc. Fourth ACM/IEEE Int'l Conf. on Human-Robot Interaction (HRI'09)*, 2009, pp. 37–44.

[14] A. Kelly, N. Chan, H. Herman, D. Huber, R. Meyers, P. Rander, R. Warner, J. Ziglar, and E. Capstick, "Real-time photorealistic virtualized reality interface for remote mobile robot control," *Int'l J. of Robotics Research*, vol. 30, no. 3, pp. 384–404, 2011.

[15] B. Hine, P. Hontalas, T. Fong, L. Piguet, E. Nygren, and A. Kline, "VEVI: A virtual environment teleoperations interface for planetary exploration," in *Proc. 25th SAE Int'l Conf. on Environmental Systems*, 1995.

[16] L. A. Nguyen, M. Bualat, L. J. Edwards, L. Flueckiger, C. Neveu, K. Schwehr, M. D. Wagner, and E. Zbinden, "Virtual reality interfaces for visualization and control of remote vehicles," *Autonomous Robots*, vol. 11, no. 1, pp. 59–68, 2001.

[17] K. Yamazawa, Y. Yagi, and M. Yachida, "Omnidirectional imaging with hyperboloidal projection," in *Proc. 1993 IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS'93)*, vol. 2, 1993, pp. 1029–1034.

[18] M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, 2001.

[19] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon, "KinectFusion: Real-time 3D reconstruction and interaction using a moving depth camera," in *Proc. 24th ACM Sympo. on User Interface Software and Technology (UIST'11)*, 2011, pp. 559–568.

[20] P. Merrell, A. Akbarzadeh, L. Wang, P. Mordohai, J. M. Frahm, R. Yang, D. Nistér, and M. Pollefeys, "Real-time visibility-based fusion of depth maps," in *Proc. 11th IEEE Int'l Conf. on Computer Vision (ICCV'07)*, 2007, pp. 1–8.

[21] S. M. Seitz and C. R. Dyer, "View morphing," in *Proc. ACM SIGGRAPH'96*, 1996, pp. 21–30.

[22] M. Levoy and P. Hanrahan, "Light field rendering," in *Proc. ACM SIGGRAPH'96*, 1996, pp. 31–42.

[23] T. Naemura, T. Takano, M. Kaneko, and H. Harashima, "Ray-based creation of photo-realistic virtual world," in *Proc. Third Int'l Conf. on Virtual Systems and Multimedia (VSMM'97)*, 1997, pp. 59–68.

[24] S. B. Kang, R. Szeliski, and P. Anandan, "The geometry-image representation tradeoff for rendering," in *Proc. 2000 IEEE Int'l Conf. on Image Processing (ICIP'00)*, vol. 2, 2000, pp. 13–16.

[25] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach," in *Proc. ACM SIGGRAPH'96*, 1996, pp. 11–20.

[26] T. Sato, H. Koshizawa, and N. Yokoya, "Omnidirectional free-viewpoint rendering using a deformable 3-D mesh model," *Int'l J. of Virtual Reality*, vol. 9, no. 1, pp. 37–44, 2010.

[27] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, "Recent advances in augmented reality," *IEEE Computer Graphics and Applications*, vol. 21, no. 6, pp. 34–47, 2001.

[28] Y. Hieida, T. Suenaga, K. Takemura, J. Takamatsu, and T. Ogasawara, "Real-time scan-matching using L0-norm minimization under dynamic crowded environments," in *Proc. Fourth Workshop on Planning, Perception and Navigation for Intelligent Vehicles*, 2012, pp. 257–262.

[29] A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," in *Proc. 2009 Robotics: Science and Systems (RSS'09)*, vol. 25, 2009.