

Technical Paper:

# Offline Direct Teaching for a Robotic Manipulator in the Computational Space

Satoshi Makita<sup>\*,†</sup>, Takuya Sasaki<sup>\*\*</sup>, and Tatsuhiko Urakawa<sup>\*\*</sup>

<sup>\*</sup>Fukuoka Institute of Technology

3-30-1 Wajirohigashi, Higashi-ku, Fukuoka 811-0295, Japan

<sup>†</sup>Corresponding author, E-mail: makita@fit.ac.jp

<sup>\*\*</sup>National Institute of Technology, Sasebo College, Sasebo, Japan

[Received August 21, 2020; accepted January 18, 2021]

This paper proposes a robot teaching method using augmented and virtual reality technologies. Robot teaching is essential for robots to accomplish several tasks in industrial production. Although there are various approaches to perform motion planning for robot manipulation, robot teaching is still required for precision and reliability. Online teaching, in which a physical robot moves in the real space to obtain the desired motion, is widely performed because of its ease and reliability. However, actual robot movements are required. In contrast, offline teaching can be accomplished entirely in the computational space, and it requires constructing the robot's surroundings as computer graphic models. Additionally, planar displays do not provide sufficient information on 3D scenes. Our proposed method can be employed as offline teaching, but the operator can manipulate the robot intuitively using a head-mounted device and the specified controllers in the virtual 3D space. We demonstrate two approaches for robot teaching with augmented and virtual reality technologies and show some experimental results.

**Keywords:** robot teaching, direct teaching, augmented reality, virtual reality

## 1. Introduction

Robot teaching is an essential process to be performed to employ robots for factory automation and other manipulation tasks because the robots require the desired motion provided by the robots themselves or by humans. In flexible automation, robots are expected to adapt to various situations such as different target objects, various amounts of products and production speeds. Consequently, each desired motion of the robot also changes. Thus, quick teaching methods or highly credible motion planners are frequently required. This study focuses on robot teaching by humans because human-operated teaching appears to be faster and more flexible than automatic motion planning based on sensing, and it has also been widely employed in industrial applications for sev-

eral decades.

Current robot teaching methods can be classified into two approaches: online teaching and offline teaching. Online teaching requires a connection with physical robots to operate them directly. Thus, any production by the robots must be suspended to provide them with a new motion. A frequently used method of online teaching is *remote teaching*, where the robot is controlled by a human operator with a handy controller called a *teaching pendant*. The teaching pendant helps the operator to move the joints of the robot one by one. Hence, the operator should understand how the robot moves with the commanded joint variables.

Another online teaching method is called *direct teaching*, in which the operator physically pushes or pulls the robot's body to move it to the desired posture. This intuitive operation of robot teaching frees operators partially from calculations of robot kinematics because the robot can be moved by hand with external forces. Although the precision of positioning depends on the manipulation skill of the operator, this method facilitates robot teaching, especially for inexperienced operators. However, this type of online teaching requires continuous physical contact between the robot and the operator. Thus, the robot must stop working for a new motion.

As for offline teaching, we execute a teaching procedure on a computer without connection with physical robots. We have to prepare each computer graphic of the robots and their surroundings to reconstruct the target workspace. As every motion of the robots can be simulated only in the computational space, the physical robot is usually allowed to continue to work. However, the resolution of reconstruction is highly dependent on the quality of the preparation of computer graphics. Then, the precision of the sensing using depth sensors affects the resolution of the generated computer graphics. Therefore, the precision of robot positioning also depends on the resolution of the environment reconstructed in the computational space.

Other teaching methods as robot programming have been studied based on online and offline teaching [1]. Teaching via human demonstration helps robots intuitively understand how humans perform motions. The target motion performed by humans is observed using cam-



eras and is measured using attached sensors, such as goniometers and electromyographs. These analyzed data are transferred to the robots. Machine learning approaches are often utilized in analyses. Dillmann proposed a teaching method involving the observation of human performance [2]. Calinon and Billiard also studied robot programming via demonstration [3]. A human operator wearing motion sensors on the body performs target tasks, and a robot learns the corresponding motion from the obtained data. Maeda and Nakamura studied view-based teaching and playback for robot manipulation [4]. A camera observes an object manipulated by a human and records its trajectory. A neural network generates a corresponding robot motion to replay the trajectory of the object. Hoshino et al. observed human motion using a motion tracking system to transfer manipulation skills to a teleoperated robot [5].

Motion planning can be regarded as a type of offline teaching. Finding the paths of robots has been widely studied for a long time [6]. Both the aforementioned robot programming methods and motion planning approaches are useful for automatically generating robot motion for manipulation. However, robot teaching by humans is more suitable for several cases on the production line compared with motion planning from precision and reliability.

This paper first proposes a method of robot teaching using augmented reality (or mixed reality). A human operator wears a head-mounted device and views computer graphics laid over the real world. An advantage of online teaching, particularly direct teaching, is that the operator intuitively moves robots with physical contact. However, the robots must be moved to obtain the desired motion in the real world. In contrast, offline teaching can be performed entirely in the computational space without any physical robot operation. Our proposed idea for robot teaching is to perform the direct teaching of robots in the computational space. The head-mounted device with a display provides the operator with a 360-degree view similar to the real space. The operator can manipulate the computer graphics of the robots intuitively using hand gestures recognized using depth sensors or specified controllers for the virtual reality system. Therefore our proposed method accomplishes offline robot teaching with an intuitive operation of robot motion.

Gadre previously proposed robot teaching via mixed reality, where the operator uses hand gestures and voice inputs to specify each subgoal of robot motion [7]. Rosen et al. also attempted communicating and controlling robot arm motion via an augmented reality device [8]. They visualized the planned robot arm motion on the head-mounted display and adjusted the goal posture of the arm using hand gestures tracked by the device. However, such a teaching system using mixed reality devices is restricted in several situations. For example, the operator cannot approach the robot sufficiently closely. In this case, a virtual reality space constructed with the scanned surroundings around the robot may be useful. Kawasaki et al. proposed a robot teaching method in a virtual environment through



**Fig. 1.** Screenshot of the see-through display, where a computer graphic of the robot is laid over the physical robot.

bimanual multi-fingered haptic interfaces [9].

Robot teaching and teleoperation with augmented and virtual reality technologies have been applied to not only manipulators but also mobile robots. Kato et al. proposed a teleoperation system with force feedback and a see-through display to control mobile robots remotely [10]. Hoshino et al. also focused on force and visual feedback with virtual reality to operate a legged robot using haptic controllers [11]. We focus on robotic manipulators and propose an intuitive offline teaching method with both augmented and virtual reality technologies. The remainder of this paper describes our proposed methods and demonstrates their performances.

## 2. Teaching Procedure with Augmented/Virtual Reality Devices

There are two cases of environments around the target robot for robot teaching via augmented or virtual reality: A) the operator who provides the robot with a motion for tasks can approach the robot sufficiently closely so that he/she can touch the robot, and B) the operator stands far from the robot. The former case is similar to usual physical direct teaching, and the operator can view the robot and its surroundings. In this situation, a computer graphic model (called simply *a model*) of the target robot can be laid over the physical robot when a see-through display is used to view both the robot and its model simultaneously (**Fig. 1**). The latter case is almost the same as teleoperation, in which the operator is usually far from the robot owing to fences. In this situation, it is difficult to view both the robot and its model via augmented reality. Thus, we have to teach the robot in the computational space. We use augmented reality to avoid constructing computer graphics of the surroundings around the robots. When this situation cannot be avoided, we construct the surround-

ings in the computational space to perform offline direct teaching via virtual reality.

The details of each method can be summarized as follows.

#### Using augmented reality

When the operator wears a head-mounted computer with a see-through display, the computer graphic model of the target robot can be laid over the physical robot to allow the operator to view both the computer graphic and the physical robot simultaneously. It is not necessary to construct other computer graphics, such as the surroundings of the robot. In our experiments, we use a head-mounted computer with a see-through display, Microsoft HoloLens, which was developed explicitly for augmented reality technology, known as *mixed reality*. The device also has depth cameras to measure the distance from obstacles around the operator who wears the head-mounted device. The depth images of the surroundings acquired using the depth cameras are mapped in the computational space, and the location of the device is also estimated. In addition, HoloLens can recognize several hand gestures, such as pinching motions called *air tap* and tap-and-hold gestures [12].

#### Using virtual reality

The virtual reality system is fully demonstrated in the computational space. In the coordinates of the space, the location of each object is arbitrarily set. To arrange the computer graphics of the surroundings in the real space, we have to construct these graphics one by one and determine their relative positions and orientations. Some scanning methods, such as simultaneously localization and mapping (SLAM) [13], using depth cameras or laser range finders can aid the construction.

### 2.1. Robot Teaching with the Augmented Reality Device

The proposed robot teaching procedure with augmented reality is described as follows.

- Step 1.** The operator wears a head-mounted computer with a see-through display and stands near the robot.
- Step 2.** The depth cameras attached to the device scan the surroundings of the device, and then the device estimates its relative location in the space.
- Step 3.** The operator determines the base position of the computer graphic model of the robot to mount it. There are two ways to determine the position: manually or by scanning augmented reality markers.
- Step 4.** After the determination of the base position, a model of the robot and a spherical marker appear. The operator can manipulate the marker by hand gestures whose recognition is built in the device, HoloLens.

**Step 5.** The robot model changes its posture to chase the marker, where each joint angle of the robot is calculated by solving inverse kinematics.

**Step 6.** When the robot model reaches the appropriate position and orientation, the operator can save its posture as a waypoint of robot motion. The operator iterates the above procedure of setting waypoints to provide the robot motion.

#### 2.1.1. Localization and Mapping

The positioning of the computer graphics of the robot on the see-through display should be matched with that of the physical robot in the real space to synchronize both motions. An SLAM method using the depth cameras attached to the device is employed to estimate the relative posture between the device and its surroundings. The relative posture is reflected in the visualization of the robot model on display. First, the depth cameras scan the surroundings around the device and generate the mapping data of the real space. As the scanning is iterated, the scanned depth images are collated with the generated map on each occasion. Finally, the relative position and orientation of the device are estimated.

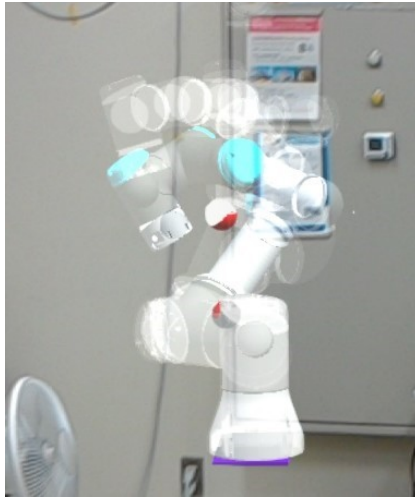
#### 2.1.2. Fixing the Base Position of the Robot

The base position of the robot is the location at which the computer model of the robot is mounted in the computational space constructed from the scanned map. We propose two methods to fix the base position: dragging an anchor manually and using augmented reality markers. For dragging an anchor manually, the operator wearing the head-mounted device can perform hand gestures to manipulate a computer graphic model of the anchor in the computational space. After the anchor model is fixed at a certain posture related to the mapping structure, the robot model is mounted at the base position.

When we use augmented reality markers, the cameras of the head-mounted device detect the markers and estimate their posture. In our experiments, we adopt Vuforia markers [14] generated with arbitrary images. The posture of the marker in the real world is estimated via image processing with the device. As a reference point of the robot, the posture of the marker corresponds to the robot model. Thus after the posture of the marker is estimated, the robot model can be arranged in the computational space by being laid over the real space on the see-through display. In this study, we adopt the Mixed Reality Toolkit (MRTK [12]) and Vuforia Engine to recognize augmented reality markers by the head-mounted device.

#### 2.1.3. Specifying the Posture of the Robot Model

For our robot teaching, we set some waypoints of the robot, which represent each desired posture in the entire robot motion. Then, the operator wearing the head-mounted device manipulates a graphical marker shown on the see-through display using hand gestures. The marker represents the desired position of the endpoint of the robot



**Fig. 2.** Example of teaching results using augmented reality. Red sphere: markers representing the desired position of the robot model.

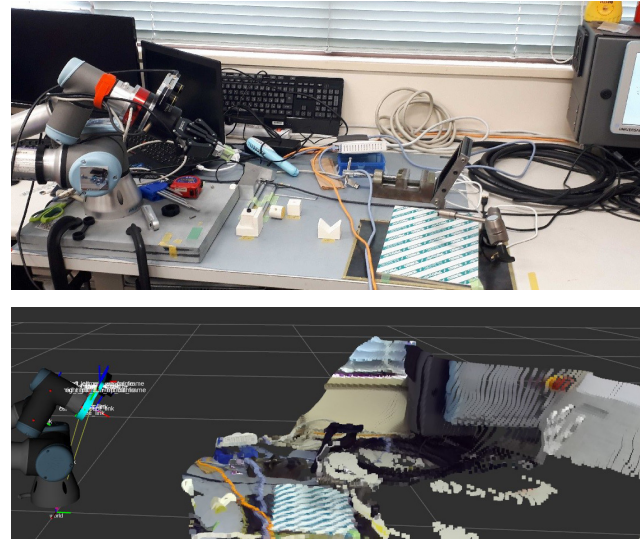
model. When the manipulator can reach the marker, the position of the marker is recorded, and each joint variable of the robot is calculated by solving inverse kinematics. Otherwise, no values are recorded and calculated. The operator iterates the following procedure to specify each posture of the robot model.

- Step 1.** The marker appears in the computational space and changes its color when the operator selects it to manipulate it using a hand gestures.
- Step 2.** The colored marker is dragged freely by another hand gesture as the desired position of the end-point of the manipulator.
- Step 3.** The computer graphic of the robot dynamically changes its posture by changing each joint variable.
- Step 4.** When the marker reaches the desired location, the teaching procedure is completed, and the operator can add another marker for the next waypoint.

A drawing function to illustrate each waypoint and the robot motion path is implemented, as shown in **Fig. 2**. This helps the operator check both the current and previous paths. Linear interpolation between each waypoint in the joint space is employed to generate a path containing all the waypoints.

## 2.2. Robot Teaching with the Virtual Reality Device

When the operator cannot approach the robot, the aforementioned method with augmented reality is not suitable because the operator cannot view the physical robot and its surroundings using the see-through display. In this case, we attach a depth camera to the robot's wrist to obtain the scanned data of its surroundings to construct the computer graphics in the computational space (**Fig. 3**).



**Fig. 3.** Upper: color image around the robot in the real space. Lower: point cloud of the scanned map from the camera attached to the robot.

### 2.2.1. Construction of Scanned Map Around the Physical Robot

Some assumptions have been made to construct scanned and mapped surroundings in the computational space. The base position of the robot model in the computational space is given in advance. Moreover, the posture of the camera on the robot is established. The procedure of scanning and mapping the surroundings around the robot is described as follows.

- Step 1.** The depth camera attached to the robot captures color and depth images. The scanned data are automatically stored as a point cloud.
- Step 2.** The operator can control the posture of the robot using the method described in Section 2.1.3 to scan a wider area.
- Step 3.** The OctoMap method compresses the data size of the point cloud to reduce data-processing costs.

Generally, the point cloud data are large because each point has its coordinates in the image, RGB color, and the distance from the camera. Hence, we compress the data using OctoMap to avoid high computational and traffic costs to process the entire data of the point cloud. OctoMap is a probabilistic 3D mapping framework based on Octree [15, 16], and it generates an approximated map of the surroundings. Although the approximation reduces the precision of the scanned map, the data size of the map can be also reduced.

### 2.2.2. Specifying the Posture of the Robot Model

As in Section 2.1.3, the method of virtual direct teaching in the fully computational space (virtual reality) is similar to that of augmented reality. The operator can



view both the computer graphics of the robot and its surroundings through the display on the head-mounted virtual reality device. A marker as the desired position of the endpoint of the manipulator appears in the scene, and it is manipulated using input given by the operator with a hand-held controller. Viewpoints provided for the operator can be arbitrarily changed to enlarge or downsize a particular scene, such as a target object. When the operated robot model occupies other objects and surroundings, collision detection can occur, and it can be indicated to the operator.

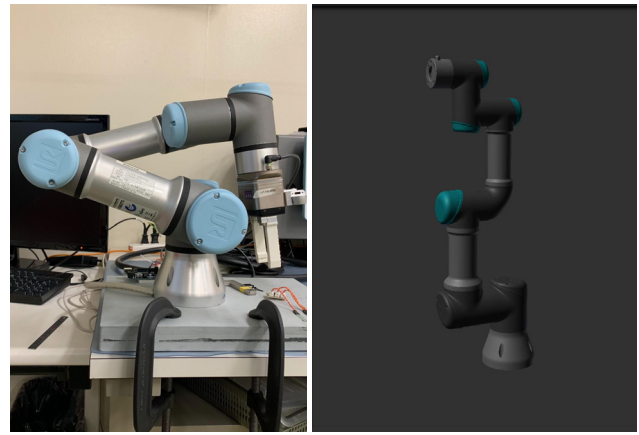
### 3. Experimental Results

In this paper, we propose methods for offline direct teaching using augmented or virtual reality technologies. These two approaches can be selected to determine whether the operator is sufficiently close to the robot to view the details of the physical robot. When the operator uses augmented reality, a computer graphic of the robot can be viewed on the see-through display built in the head-mounted computer, and it can be manipulated using hand gestures. Otherwise, the depth camera attached to the endpoint of the manipulator obtains a point cloud of its surroundings, and the scanned data based on the point cloud are mapped in the computational space to reconstruct the surroundings virtually. Thus, the precision of this mapping of the surroundings is essential for our proposed method utilized instead of augmented reality. Experiments are conducted by focusing on each objectives: for virtual direct teaching with augmented reality, whether the proposed method is applicable as a supplement to physical direct teaching, which is widely utilized in industrial scenario; for the method using the virtual reality space, whether the accuracy of perception of the target object based on the mapping with the depth camera is sufficient.

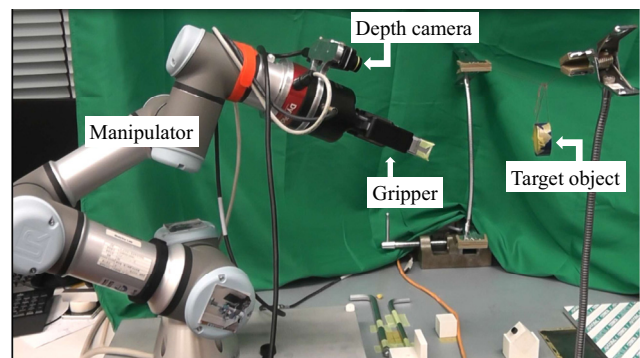
#### 3.1. Experimental Setup

Our proposed methods are selected according to whether the operator can approach the robot sufficiently closely. The robot manipulator commonly used in the experiments is UR3-CB (Universal Robots, Inc.) and is operated manually and commanded by ROS [17, 18]. The ROS system is a framework of robot development with robots, sensors, and cameras and provides developers with programming libraries for those devices, motion planners, visualization, and other useful tools. The 3D CAD data of the robot were officially provided (Fig. 4). Hence, we can translate the data into the appropriate format and add each joint setting of the robot to introduce to the Unity framework (Unity, Inc. [19]), a development framework of 2D/3D games. We generate 360-degree views with the game engine for both teaching head-mounted devices.

For the teaching system with augmented reality, we adopted Microsoft HoloLens (Microsoft Inc.), which has



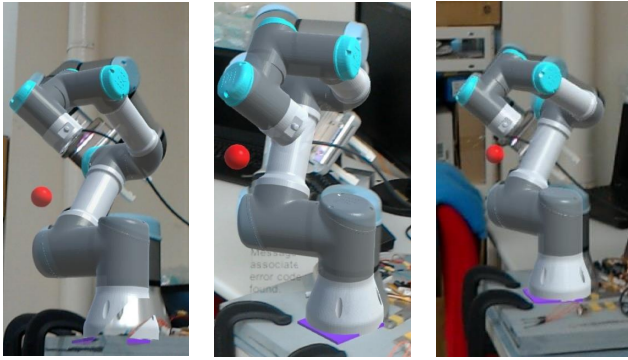
**Fig. 4.** Physical robot (left) and its computer graphic model (right).



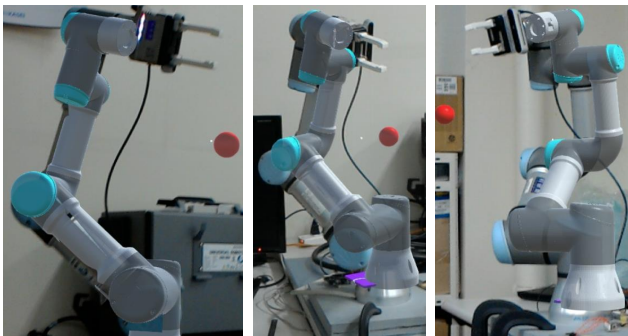
**Fig. 5.** Experimental setup for the teaching system with virtual reality. For the method with augmented reality, we remove the depth camera.

a see-through display in front of the operator's eyes. Hence, the operator can look around the real space with some displayed information. The stand alone device has depth cameras to obtain a scanned map around the device and processes the map for the localization of the device. As the recognition of hand gestures is also provided, the operator can interact with the displayed objects using several hand gestures as the user's inputs. Additionally, we adopted the Vuforia Engine [14] as a reader of augmented reality markers, which operates in the Unity framework. The augmented reality markers are attached to the body of the robot and is scanned by the camera on the head-mounted device to estimate the posture of the marker.

For teaching with virtual reality, we adopted Oculus Rift S (Oculus, Inc.), a head-mounted display with two hand-held controllers. The controllers are tracked to reflect the posture of the operator's hands as the user's inputs to the Unity framework. As the depth camera for scanning around the robot, a stereo vision camera, ZED Mini (Stereolabs, Inc.), is attached to the wrist of the robot (Fig. 5). The depth camera is driven by ROS to acquire both color and depth images, and image processing including the OctoMap method is also executed in the ROS framework.



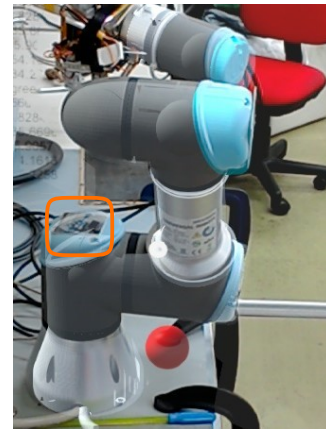
**Fig. 6.** Teaching result with augmented reality. The computer graphic with joints colored with deeper green-blue is laid over the physical robot with joints colored with light blue. The computer graphic expresses the posture given by the operator in the computational space in augmented reality. We can view the physical robot, which moves according to the posture, behind the computer graphic because there is a misalignment between them.



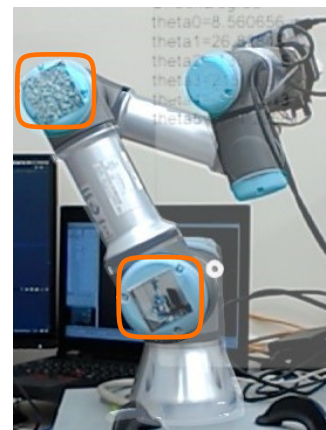
**Fig. 7.** Another result with augmented reality. The computer graphic with joints colored with deeper green-blue does not have a gripper, whereas the physical robot with joints colored with light blue has a robotic gripper. We can view a misalignment between both postures attributed to the misalignment of both base positions.

### 3.2. Results of Robot Teaching with Augmented Reality

As explained in Section 2.1, first, the operator who wears a head-mounted computer with a see-through display, HoloLens, determines the base position of the manipulator, and subsequently teaches the waypoints of the end position of the manipulator by manually manipulating the graphic of the robot that appears. In these experiments, each joint variable is calculated according to the waypoints by solving inverse kinematics and is provided to the physical robot manually by using a teaching pendant. The results of teaching with augmented reality are shown in **Figs. 6** and **7**. According to the results, there is a misalignment between the manipulated computer model and the physical robot. The misalignment is caused by the manually determined base position of the robot model. When the operator determines the base position, he/she visually adjusts the graphic of its anchor to the base part



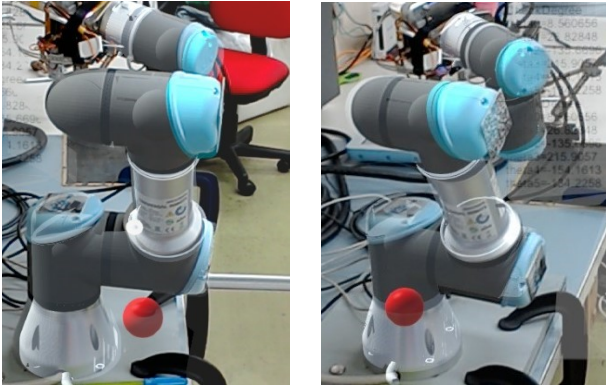
**Fig. 8.** Augmented reality marker attached to the robot in the horizontal direction.



**Fig. 9.** Augmented reality markers attached to the robot in the vertical direction.

of the physical robot. Thus, the precision of robot positioning decreases. On the other hand, as the joint variables of both the computer graphic and the physical robot are equal, each posture of the robot appears to be the same. Thus, we introduced the detection of augmented reality markers to avoid a misalignment of the base position.

The misalignment of the robot caused by manual setting can be fairly avoided using augmented reality markers detection powered by the Vuforia Engine [14]. The augmented reality marker is attached to the first joint of the robot that is perpendicular to the rotation axis of the joint (**Fig. 8**) because this side is always horizontal regardless of every joint variable. The misalignment can be reduced with more markers. Another augmented reality marker attached to the third joint that is perpendicular to the rotation axis of the joint, which is always vertical, contributes to reducing the horizontal misalignment (**Fig. 9**). Thus, the combination of the augmented reality markers, which is attached to the side of the robot in the horizontal and vertical direction, is effective to avoid the fundamental misalignment between the robot model and the physical robot (**Fig. 10**). Note that the specifications of the computer graphic of the robot are almost the same



**Fig. 10.** Teaching results with the augmented reality markers. As in Fig. 6, the computer graphic with joints colored with deeper green-blue is laid over the physical robot with joints colored with light blue. Then we can view less misalignment between them because recognition of augmented reality markers improves the precision of positioning of the base of the robot.

as those of the physical robot because they are based on the CAD data of the robot. Additionally, each joint variable is equal. Therefore, every misalignment in the teaching results strongly depends on the teaching skills of the operator, similar to (physical) direct teaching.

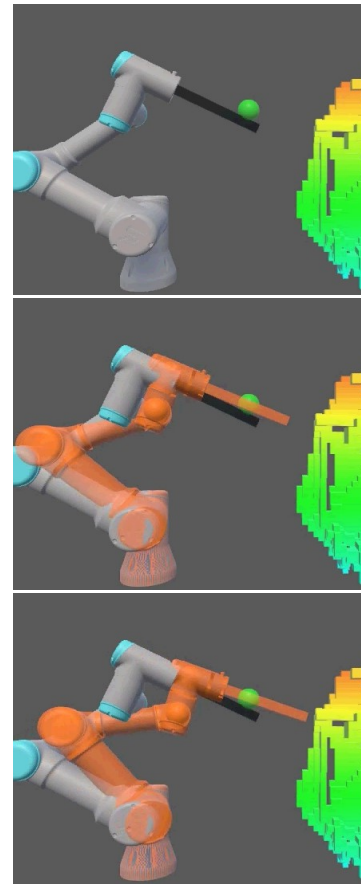
### 3.3. Results of Robot Teaching with Virtual Reality

When we do not apply the teaching method with augmented reality due to situations involving inaccessibility of the robot, we adopt the virtual reality approach. In this case we have to build each component of the surroundings around the robot and the target objects. First, we scan the surroundings with a depth camera and then, manipulate a robot model in the mapped computational space to teach the desired motion. The remainder of this section describes the details of the procedure and the experimental results.

#### 3.3.1. Experimental Procedure

We demonstrated robot teaching for a reaching motion with the following procedure to investigate the precision of the surroundings in the computational space reconstructed from the scanned and mapped data of the depth camera.

- Step 1.** Map the point cloud in the computational space, which is acquired by the depth camera on the robot.
- Step 2.** Approximate the point cloud with larger voxels to reduce the amount of the scanned data.
- Step 3.** Estimate the posture of the target object with the approximated voxels.
- Step 4.** Manually operate the computer graphics of the robot using hand-held controllers so that the end effector of the robot reaches the target.



**Fig. 11.** Result of the virtual direct teaching of the manipulator. The operator manipulates the green spherical marker representing the desired position of the end tool of the robot model. When the operator determines the location of the marker, the robot model obtains each joint variable by solving inverse kinematics. The orange manipulator represents the desired posture of the robot model.

**Step 5.** Calculate each joint variable of the manipulator by solving inverse kinematics.

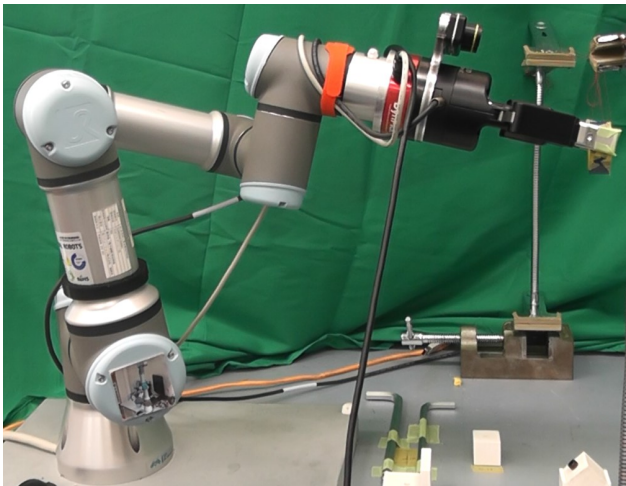
**Step 6.** Control the physical manipulator with the obtained joint variables as a teaching playback.

A robotic gripper with two adaptive fingers, 2F-85 (Robotiq Inc.), is attached to the manipulator as an end effector. The target object is a cylinder with a diameter of 45 mm and a height of 19 mm, and is hung as shown in Fig. 5.

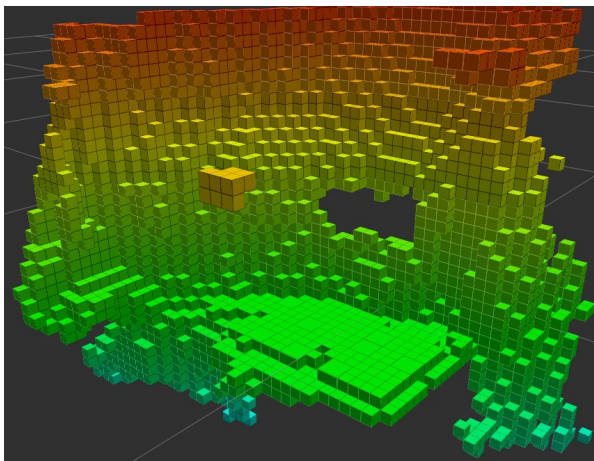
#### 3.3.2. Experimental Results of Robot Teaching with Virtual Reality

Consequently, the robotic gripper successfully reached the target object (Figs. 11 and 12). The estimated  $(x, y, z)$  coordinates of the center of the target object in the computational space are  $(0, 0.378, 0.5)$  [m], and the manually given goal position of the robotic gripper is  $(0.005, 0.3779, 0.4949)$  [m]. Thus, each relative error of positioning is acceptable for the teaching of grasping tasks.

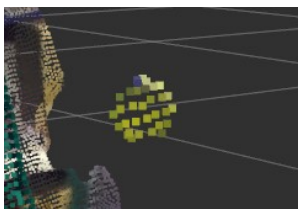




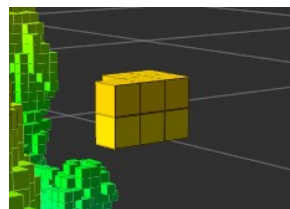
**Fig. 12.** Result of reaching motion provided by the proposed teaching method in the computational space (Fig. 11).



(a) Scanned and approximated object and surroundings in the scene of Fig. 5



(b) A point cloud of the target object scanned using the depth camera



(c) Approximation of the scanned target object using OctoMap

**Fig. 13.** Target object and its surroundings scanned using the depth camera and approximated by voxels.

Note that we adopted OctoMap, an approximation with voxels, in the ROS framework to reduce the amount of data transferred to the Unity framework as the mapped surroundings (Fig. 13(a)). This approximation causes a loss of description of the target object. In the OctoMap processing, the entire space is divided into voxels with a size of  $0.02 \text{ m}^3$ . A voxel is selected as solid when the parts of the point cloud are contained in the voxel. Otherwise

a voxel is selected as empty. Hence, the object approximated by the voxels appears larger than the real one in general. In the experiments, the approximated object consists of eight voxels and has a volume of  $6.4 \times 10^{-5} \text{ m}^3$ , whereas the volume of the real object is  $3.02 \times 10^{-5} \text{ m}^3$  (Figs. 13(b) and (c)). Although the difference in the volumes of the target object is not small, the object coordinates can be fairly estimated.

Since the operator can easily change viewpoints in the computational space to enlarge or downsize the computer graphics of the object, he/she can precisely move the robot model to allow it to reach the target object. On the other hand, the target object in our experiments is still hung and is free from other obstacles. If the object is on a flat surface, the approximated object and the other obstacles must be in contact with each other. In this situation, it becomes difficult to separate them and to estimate the precise coordinates of the target object. Additionally, the computer graphic of the robot easily occupies those of the other obstacles while the operator manipulates it with the virtual reality controllers. Thus, to avoid such problems, the size of the voxel should be smaller to express the scanned surroundings with a high resolution, although the amount of data to be transferred increases. In such cases, we also have to implement a collision detection method and indicate it to the operator.

## 4. Conclusions

This paper presented a method of offline direct teaching for robotic manipulators using augmented and virtual reality. The operator moves the computer graphics of the target robot on the head-mounted display intuitively using hand gestures. The differences between the two situations and our contributions are summarized as follows.

### Robot teaching with augmented reality

A human operator wearing a specified head-mounted computer with a see-through display can view a computer graphic model of a target robot on display. The robot model was laid over the physical robot in real space. Thus, computer graphics for the surroundings need not be prepared. The operator manipulates a marker graphic representing the desired position of the endpoint of the robot model using hand gestures. The above operation can determine each waypoint on the path of the robot's motion.

### Robot teaching with virtual reality

A human operator wearing a specified head-mounted display views computer graphics; hence, the construction of entire surroundings is required. A depth camera attached to the robot arm scans the surroundings around the robot and maps the scanned data in the computational space. Then the scanned map is approximated with voxels to reduce the transferred data. The operator manipulates a marker representing the desired position of the robot using the hand-held controllers in the virtual space. Subsequently, the physical robot demonstrated a reaching motion according to the robot teaching.



In future works, the proposed method will be applied to various production and manipulation situations, and its versatility will be investigated. Additionally, the resolution of mapped surroundings in the computational space will be improved, which may be affected by scanning resolution and data processing speed.

### Acknowledgements

This work was supported by JSPS KAKENHI Grant Number JP 17H04699.

### References:

- [1] Z. Pan, J. Polden, N. Larkin, S. V. Duin, and J. Norrish, "Recent progress on programming methods for industrial robots," *Robotics and Computer-Integrated Manufacturing*, Vol.28, No.2, pp. 87-94, 2012.
- [2] R. Dillmann, "Teaching and learning of robot tasks via observation of human performance," *Robotics and Autonomous System*, Vol.47, pp. 109-116, 2004.
- [3] S. Calinon and A. Billard, "Active Teaching in Robot Programming by Demonstration," *Proc. of the IEEE Int. Symp. on Robot and Human Interactive Communication*, pp. 702-707, 2007.
- [4] Y. Maeda and T. Nakamura, "View-based teaching/playback for robotic manipulation," *Robomech J.*, Vol.2, 2, 2015.
- [5] K. Hoshino, N. Igo, M. Tomida, and H. Kotani, "Teleoperating System for Manipulating a Moon Exploring Robot on the Earth," *Int. J. Automation Technol.*, Vol.11, No.3, pp. 433-441, 2017.
- [6] J.-C. Latombe, "Motion Planning: A Journey of Robots, Molecules, Digital Actors, and Other Artifacts," *Int. J. of Robotics Research*, Vol.18, No.11, pp. 1119-1128, 1999.
- [7] S. Y. Gadre, "Teaching Robots Using Mixed Reality," Master's thesis, Brown University, 2018.
- [8] E. Rosen, D. Whitney, E. Phillips, G. Chien, J. Tompkin, G. Konidaris, and S. Tellex, "Communicating and controlling robot arm motion intent through mixed-reality head-mounted displays," *Int. J. of Robotics Research*, Vol.38, Nos.12-13, pp. 1513-1526, 2019.
- [9] H. Kawasaki, T. Mouri, and S. Ueki, "Virtual Robot Teaching for Humanoid Both-Hands Robots Using Multi-Fingered Haptic Interface," C. S. Lanyi (Ed.), "The Thousand Faces of Virtual Reality," pp. 107-128, doi: 10.5772/59189, 2014.
- [10] K. Kato, N. Sato, and Y. Morita, "Development of direct operation system for mobile robot by using 3D CG diorama," *Proc. of the Int. Conf. on Control, Automation and Systems*, pp. 1486-1490, 2012.
- [11] K. Hoshino, M. Kitani, R. Asami, N. Sato, Y. Morita, T. Fujiwara, T. Endo, and F. Matsuno, "Improvement of operability of teleoperation system for legged rescue robot," *Proc. of the IEEE Int. Conf. on Intelligence and Safety for Robotics*, pp. 134-139, 2018.
- [12] Microsoft, "Mixed Reality Toolkit." <https://microsoft.github.io/MixedRealityToolkit-Unity/README.html> [Accessed August 8, 2020]
- [13] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics and Automation Magazine*, Vol.13, No.2, pp. 99-110, 2006.
- [14] PTC, "Vuforia Engine." <https://developer.vuforia.com/> [Accessed August 8, 2020]
- [15] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees," *Autonomous Robots*, Vol.34, pp. 189-206, 2013.
- [16] "Octomap." <https://octomap.github.io/> [Accessed August 8, 2020]
- [17] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," *ICRA Workshop on Open Source Software*, 2009.
- [18] "ROS: The Robot Operating System." <https://www.ros.org/> [Accessed August 8, 2020]
- [19] Unity Technologies, "Unity." <https://unity.com/> [Accessed August 8, 2020]

**Name:**

Satoshi Makita

**Affiliation:**

Department of Intelligent Mechanical Engineering, Faculty of Engineering, Fukuoka Institute of Technology

**Address:**

3-30-1 Wajirohigashi, Higashi-ku, Fukuoka 811-0295, Japan

**Brief Biographical History:**

2010-2020 Faculty Member, National Institute of Technology, Sasebo College

2020- Assistant Professor, Fukuoka Institute of Technology

**Main Works:**

- S. Makita and W. Wan, "A survey of robotic caging and its applications," *Advanced Robotics*, Vol.31, Issues 19-20, pp. 1071-1085, 2017.
- S. Makita and Y. Maeda, "3D Multifingered Caging: Basic Formulation and Planning," *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2697-2702, 2008.

**Membership in Academic Societies:**

- Robotics Society of Japan (RSJ)
- Japan Society of Mechanical Engineers (JSME)
- Institute of Electrical and Electronics Engineers (IEEE)

**Name:**

Takuya Sasaki

**Affiliation:**

Advanced Course of Integrated Engineering, National Institute of Technology, Sasebo College

**Address:**

1-1 Okishinmachi, Sasebo, Nagasaki 857-1193, Japan

**Brief Biographical History:**

2019- Undergraduate Student, Advanced Course of Integrated Engineering, National Institute of Technology, Sasebo College

**Main Works:**

- Mixed reality, robot programming, and AI player in computer games

**Membership in Academic Societies:**

- Japan Society of Mechanical Engineers (JSME)

**Name:**

Tatsuhiko Urakawa

**Affiliation:**

Department of Control Engineering, National Institute of Technology, Sasebo College

**Address:**

1-1 Okishinmachi, Sasebo, Nagasaki 857-1193, Japan

**Brief Biographical History:**

2015-2020 Undergraduate Student, Department of Control Engineering, National Institute of Technology, Sasebo College

**Main Works:**

- Virtual reality and robot programming