We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter 6

Virtual Robot Teaching for Humanoid Both-Hands Robots Using Multi-Fingered Haptic Interface

Haruhisa Kawasaki, Tetsuya Mouri and Satoshi Ueki

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/59189

1. Introduction

Humanoid robots are expected to support various human tasks, such as high-mix low-volume production and assembly tasks. Programming technologies such as robot language and teaching-playback [1] have been developed, but it is difficult to apply these technologies to multi-fingered robots because they require instruction for both motion and force at many points simultaneously.

Recently, several techniques have been proposed that use human motion measurements directly as robot teaching data for automatic programming: teaching by showing [2], assembly plan from observation [3-4], gesture-based programming [5], and robot learning [6-10]. Applications for dual arm robots [11] have also been presented. These are based on measurements of motions and forces generated in the real world. Task programming based on the observation of human operation is viable for humanoid robots because it is not necessary to describe motions and forces explicitly for the robot to accomplish a task.

Direct teaching that involves remote robot operation [12] presents two difficulties. The first is caused by the communication time lag that arises when the robot is distant from the operator, which can make the remote-robot system unstable. The second is the issue of constant operator stress when any mistake on the operator's part is immediately reflected in the robot motions and coudl result in a fatal accident. Robot teaching in a virtual reality (VR) environment can overcome these problems. We call this approach VR robot teaching. Several approaches to analyzing human intentions from human demonstrations have been presented [13-17]. Most of these studies, however, do not handle the virtual force generated in the VR environment as robot teaching data. Moreover, research on VR robot teaching for humanoid both-hands robots has not yet been presented.

INTECH
open science | open minds

© 2014 The Author(s). Licensee InTech. This chapter is distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and eproduction in any medium, provided the original work is properly cited.

Our group presented a concept of VR robot teaching for multi-fingered robots [18] in which the virtual forces at contact points were utilized to analyze human intention. We found that humans feel comfortable handling virtual hands based on a human hand model, but uneasy handling virtual hands based on a robot hand model, because the geometrical form and motional function of the robot hand is not the same as that of the human hand. To minimize the difference between human-robot fingertip position and orientation, mapping methods from human grasps to robot grasps have been studied [19-20]; these did not, however, take the manipulability of the robot hand into consideration. Hand manipulability is a key measure for stable and robust robot grasps [21]. Moreover, a segmentation method for processing human motion data, including plural tasks, whose segmentation tree is additive for any new primitive motion required for performing a new task, has been presented [22]. These studies did not, however, handle 3D forces at contact points because a force-feedback glove, consisting of a data glove and a force display mechanism using wire rope, were used as a haptic interface, which could only display one dimensional force to the fingertips. Hence, it was difficult to teach a task that included a contact with another object and a motion along a surface of the object. 3D forces at contact points are key information for human intention analysis. Moreover, an expansion of VR robot teaching is needed for humanoid both-hands robots.

This paper presents a VR robot teaching method [23] for a humanoid robot hand using a multi-fingered haptic interface capable of displaying 3D force at each fingertip of the operator. In this teaching, segmentation of motion, task recognition, and re-segmentation of motion are executed sequentially using 3D forces. That is, human motion data consisting of contact points, grasped force, hand and object positions, and the like are segmented into plural primitive motions based on a segmentation tree: the type of task is analyzed based on the acquired sequence of primitive motions, and re-segmentation of the motion is executed sequentially. We demonstrate how the segmentation tree is additive for new primitive motions as part of performing a new task using 3D forces. In this method, the position and orientation of the robot hand are determined so as to maximize its manipulability, on the condition that the robot grasps the object at its teaching contact point. This approach makes the virtual teaching system very user friendly. We present the experimental results of performing a task, which includes contacting another object and moving along the surface of the wall, using a humanoid robot hand named Gifu Hand III [24] and a multi-fingered haptic interface robot called the HIRO II [25]. Moreover, we extend the VR robot teaching to humanoid both-hands robots by adding primitive motions of human bimanual coordination, which consist of *Equal Grasp by both hands, Main Grasp by one hand, Pass from one hand to another,* etc., to the segmented tree. Type of task by both-hands is also analyzed based on the acquired sequence of primitive motions. This shows that the segmentation tree can also be additive for bimanual coordination tasks.

The proposed method will be useful in the efficient manufacturing of a wide variety of products in small quantities, and for teleportation with large time delay using humanoid hand robots.

http://dx.doi.org/10.5772/59189

2. Virtual robot teaching

2.1. Scheme of virtual robot teaching

A conceptual scheme of the VR robot teaching system is shown in Figure 1. The system consists of a VR robot teaching system and remotely located robot system. In the former, the human carries out various tasks in a VR environment, employing a virtual object by handling a multi-fingered haptic interface. From the series of human motions and 3D forces acting on the human's fingers, the motion intention of the human is analyzed. Based on the motion intention analysis, a series of robot commands, which include desired trajectories, grasping forces, and contact points, are generated for the object coordinate frame. The robot commands are tested in the robot simulation system and then sent to the remote robot system. In the robot system, the robot works according to the robot commands. The robot system can absorb a slight geometrical difference between the virtual space and real space, because the robot obeys commands relative to the object coordinate frame.

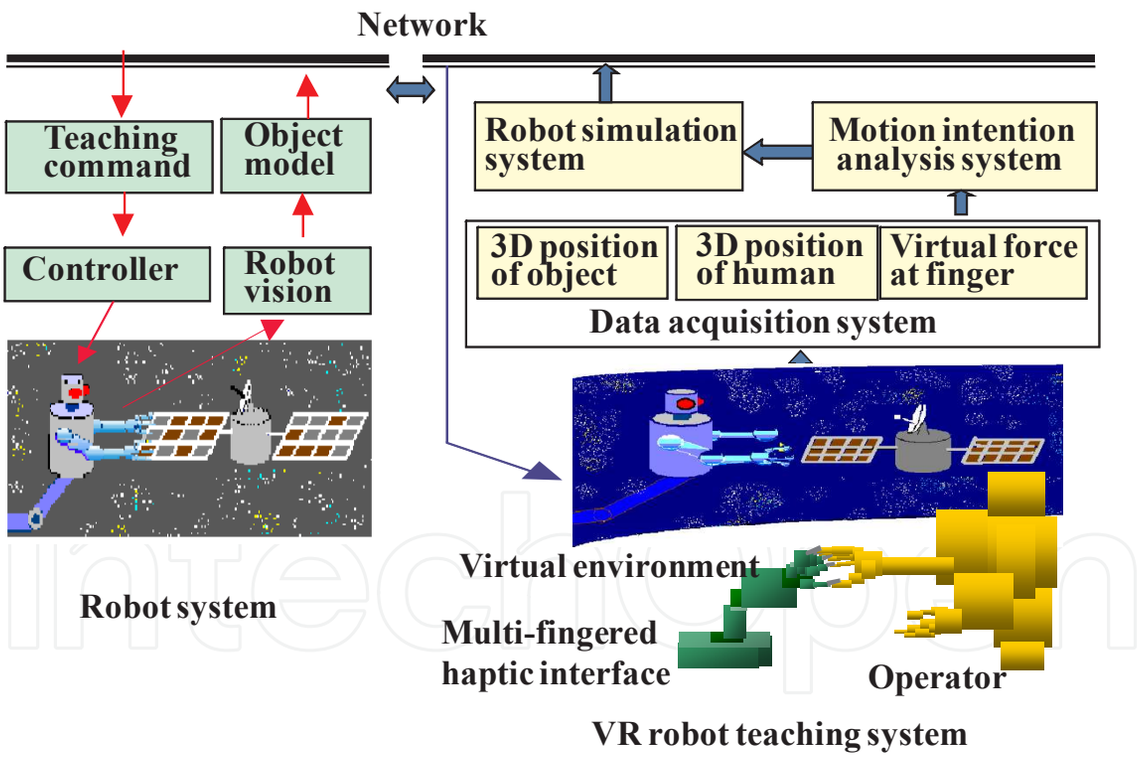


Figure 1. Conceptual figure of virtual robot teaching

This scheme has two advantages. The first is that communication time delay has no effect, since the robot commands are generated from the off-line motion analysis. The second is that the human is relieved of continual stress, since inadvertent human error can be compensated for.

2.2. Robot with Gifu Hand III

We consider a multi-fingered hand robot equipped with the Gifu Hand III developed by our group [24]. The shape and mechanism of the Gifu Hand III were designed to resemble those of the human hand. That is, it has a thumb and 4 fingers, the thumb has 4 joints with 4 degree of freedoms (DOF), and each finger has 4 joints with 3 DOF. All servomotors are mounted in the hand frame. A 6-axis force sensor can be attached to each fingertip, and a distributed tactile sensor with 859 detecting points can be mounted on the surfaces of the palm and fingers. Since the Gifu Hand III was designed so that not only the shape but also the mechanism were very similar to those of the human hand, as long as the shape of the object is simple and the size is manageable, most of the measured human motion data can be applied directly to the robot command.

2.3. Haptic interface robot HIRO II

The multi-fingered haptic interface HIRO II [25] shown in Figure 2 can present force and tactile feeling at the 5 fingertips of the human hand. The HIRO II design is completely safe. The mechanism of HIRO II consists of a 6-DOF arm and a 15-DOF hand with a thumb and 4 fingers. Each finger has 3 joints, allowing 3 DOF. The first joint, relative to the base of the hand, allows abduction/adduction. The second and third joints allow flexion/extension. The thumb is similar to the fingers except for the reduction gear ratio and the movable ranges of joints 1 and 2. In order to read the finger loading, a 6-axis force sensor is installed in the second link of each finger. The user must wear finger holders over his/her fingertips to manipulate the haptic interface. Each finger holder has a ball attached to a permanent magnet at the force sensor tip and forms a passive spherical joint. This passive spherical joint has two roles. First, differences between the human finger orientation and the haptic finger orientation are adjusted. Second, it allows operators to remove their fingers from the haptic interface in case of a malfunction. The suction force generated by the permanent magnet is 5 N. Humans can feel 3D force at each fingertip through the HIRO II.

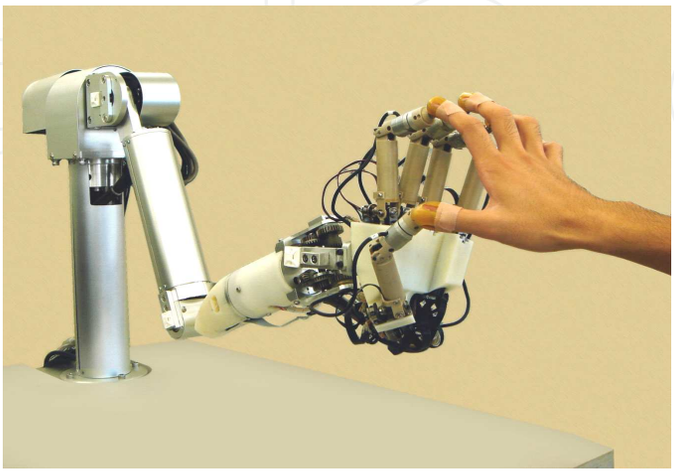


Figure 2. Multi-fingered haptic interface robot HIRO II

http://dx.doi.org/10.5772/59189

2.4. Motion segmentation

Assembly work consists of plural tasks, such as pick-and-place, peg-in-hole, peg-pullout-from-hole, and so on. Hence, the motion intention analysis system should have the ability to recognize the type of task from human motion data. A segmentation method [13] has been proposed to realize this function in which segmentation of motion, task recognition, and re-segmentation of motion are executed sequentially.

A flowchart of the segmentation of motion data is shown in Figure 3, which is a modified version of a previous one. In this segmentation tree, a pick-and-place task, peg-in-hole task, peg-pullout-from-hole task, turn-screw task, slide task, pick-and-press task, pick-and-follow task, trace task and press task are considered. The first 5 tasks can be performed without 3D force interface, but the last four cannot, because 3D contact force is required to present the human motion when the target object is in contact with multiple objects. We assumed that the human motion to execute these tasks consists of 14 primitive motions as follows: *Move*, *Approach*, *Grasp*, *Translate*, *Slide*, *Insert*, *Pullout*, *Release*, *Place*, *Turn*, *Press*, *Follow*, *Push*, and *Trace*. The last 4 primitive motions are added to segment the contact tasks when humans use the 3D force display. This means that the segmentation tree is additive for the potential use of 3D forces by a simple modification, indicated by the colored cells in Figure 3.

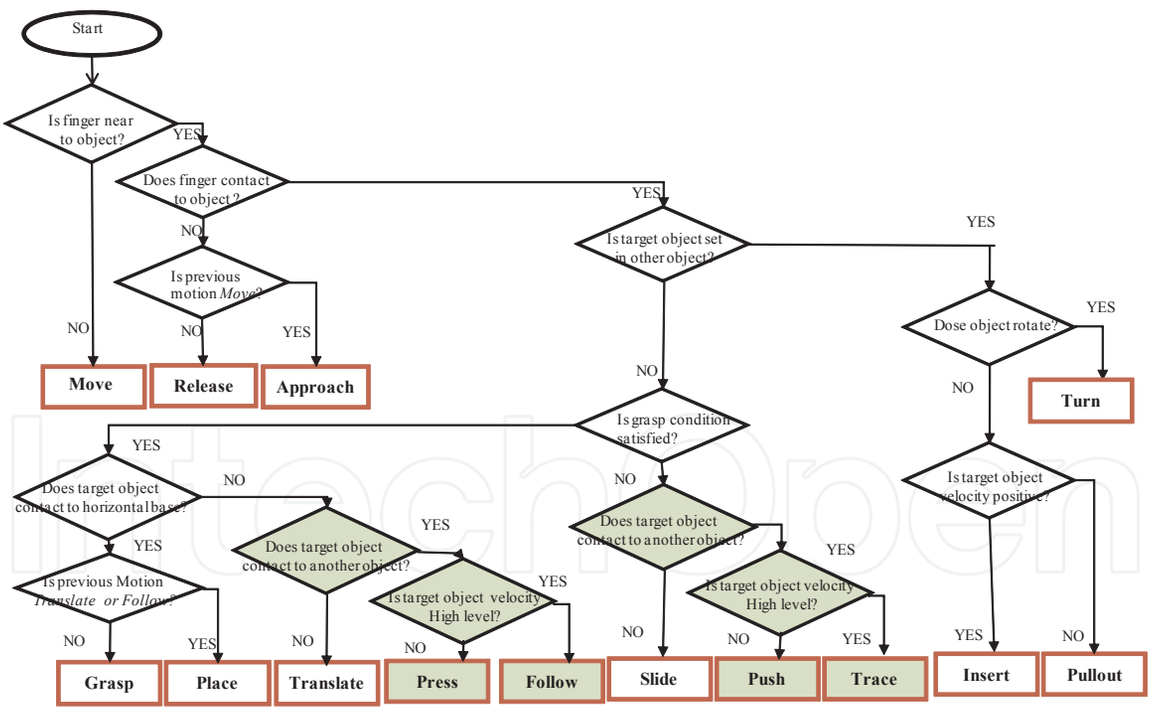


Figure 3. Flowchart of segmentation of motion data

The *Move* segment indicates only the operator's hand moving at a point distant from a virtual object; with *Approach*, the operator's fingertip is coming close to a virtual object, but does not touch it, and the previous motion is *Move*. With *Grasp*, the finger contacts an object located on a base or another object; the grasp condition is satisfied and the previous motion of the *Grasp*

is not *Translate*. The grasp condition means that the virtual force generated by the interference between the finger and virtual object is greater than a specified value. This grasp is the precision grasp presented by Cutkosky [26]. In the *Translate* segment, the object moves with the hand and fingers as one unit; that is, the virtual object departs from the base or other object and the grasp condition is satisfied. In the *Place* segment, the object contacts the environment and the previous motion is *Translate*. In human operation, it is difficult to distinguish the *Translate* and *Place* segments exactly, and indeed the operator does not feel the distinction between them. We assumed that the starting point of the *Place* segment is the moment at which the virtual object first contacts the environment. In the *Release* segment, the fingertip leaves the object and the previous motion is not *Move*. Hence, the starting point of the *Release* segment is the moment at which one of the fingertips leaves the object. In the *Slide* segment, the object is touched by the hand satisfying the grasp condition and translated to a point. In *Insert*, a finger contacts a target object set inside another object, and the target object moves toward the other object; meanwhile in *Pullout*, a finger is in contact with a target object inside another object, and the target object moves away from the other object. In the *Turn* segment, the object is turned around an axis. In the *Press* section, the object in a satisfying grasp condition contacts another object, and in *Follow,* the object in a satisfying grasp condition moves along the surface of another object with contact force. In the *Push* segment, fingers touch an object and move it toward another object until they are in contact, but the grasp condition is not satisfied. In *Trace,* fingers touch an object and move it along the surface of another object with contact force, but again the grasp condition is not satisfied.

To segment the motion data, three coordinate frames are utilized: the reference coordinate frame, the origin of which is fixed in the task space; the object coordinate frame fixed in the object; and the hand coordinate frame fixed in the hand. The following 6 parameters are measured as the motion data: an object position with respect to the reference coordinate frame ${}^{ref}\boldsymbol{p}_{object}$, an *i*-th fingertip position with respect to the object coordinate frame ${}^{object}\boldsymbol{p}_{i\text{-}th\ finger}$, a virtual force at the *i*-th fingertip with respect to the reference coordinate frame ${}^{ref}\boldsymbol{f}_{i\text{-}th\ finger}$ and the object velocity ${}^{ref}\boldsymbol{v}_{object}$. Moreover, the index of the grasp space of hand ${}^{object}\mathrm{P}_{finger}$ ($=\sum \| {}^{object}\boldsymbol{p}_{i-th\ \ finger} \|$), the sum of fingertip forces ${}^{ref}\boldsymbol{F}_{finger}$ ($=\sum {}^{ref}\boldsymbol{f}_{i-th\ \ finger}$) and the contact state flag, which indicates a contact state between the target object and the other object, are evaluated. By using these parameters, the distance between the hand and the object, the presence or absence of contact between a fingertip and the object, the contact relation between the target object and the other object, and the grasping condition are evaluated and used in the segmentation tree. As a result, the motion data is segmented into primitive motions.

2.5. Task analysis

Type of task is analyzed based on the sequence of the obtained primitive motions in the following manner. A sequence of primitive motions from *Move* to *Release* is 1 task because the hand must move to the target object to do something first, then release the object at the end of the task. When *Insert* is in the sequence of primitive motions, it is a peg-in-hole task. When the sequence of primitive motions includes *Pullout*, it is a peg-pullout-from-hole task. When *Turn* is in the sequence of primitive motions, it is a turn-screw task. When *Follow* is included in the

http://dx.doi.org/10.5772/59189

sequence of primitive motions, it is a pick-follow task. The inclusion of *Press* in the sequence of primitive motions indicates a pick-press task; *Follow* indicates a pick-and-follow task; and *Push* in the sequence of primitive motions, signifies a push task. When *Trace* and *Press* are both in the sequence of primitive motions, it is a trace task.

After task recognition, primitive motions are relabeled based on the recognized task. For example, if *Slide* is between *Place* and *Release,* it is combined into *Place,* because *Slide* is a fluctuation by the operator. Meanwhile, *Insert* and *Pullout* in a turn-screw task are combined as *Turn* because they happen at a low angular velocity to the target object. This modification process can be added as appropriate when a new primitive motion is added. After the relabeling based on the recognized task, the desired trajectories, contact points, and contact forces are analyzed within each segment.

2.6. Robot command

The geometrical form of the Gifu Hand III is similar to that of the human hand, but the two are not identical. In particular, the space between the thumb and opposing fingers is smaller in the Gifu Hand III because of a mechanical design limitation. In order to be able to map teaching data based on the human-hand model to teaching data for a robot hand, a virtual teaching method for multi-fingered robots based on a combination of scaling the virtual hand model to the size of the robot hand and hand manipulability has been developed. In this method, the position and orientation of the robot hand are determined so as to maximize the manipulability of the robot hand, on the condition that the robot grasps the object at the object's teaching contact point. More details can be found in [22].

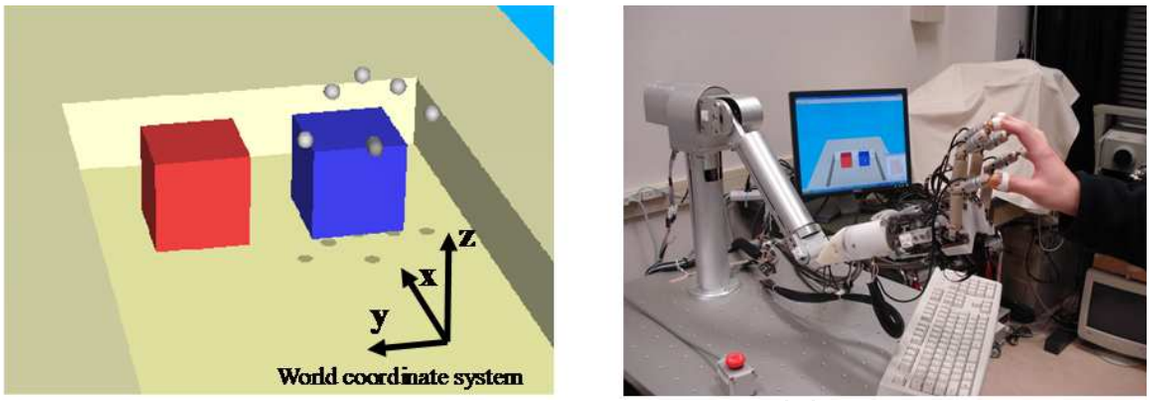
3. Experiments

3.1. Experimental system

Figure 4(a) illustrates the experimental system. An operator manipulates virtual objects in the virtual environment through the multi-fingered haptic interface robot HIRO II. There are two objects in a box; both are cubic, 120 [mm] on each side, with a mass of 50 [g] and a friction surface coefficient of 0.4. Fingertip position is indicated by a small ball in the computer graphics (CG). Static and dynamic friction coefficients between fingertip and object are 1.0 and 0.5, respectively. The robot arm is controlled by the position PID control with a friction compensator. The robot hand is controlled by the impedance control, which consists of the position PD control and the force feedback control. The control sampling cycle is 1 ms.

3.2. Pick-and-follow task

A task in which a right-side object was translated to the right-side corner was executed. The operator executed the VR robot teaching in a virtual environment, as shown in Figure 4(b). First, the operator executed a pick-and-follow task, in which he moved his hand to the target object, grasped and picked it up on the base, translated the object to the wall along the $\boldsymbol{x}$-axis



(a) Computer graphics (b) VR environment

Figure 4. Experimental system

until contact, followed along the wall to the base while grasping the object and keeping contact between the object and the wall, and then released the object. The operator then executed a push-and-trace task without grasping the object, in which he pushed the object to the wall with two fingers, traced the wall to the corner keeping contact between the object and the wall, and then released the object. Figure 5 shows the measured parameters, primitive motions obtained by the proposed segmentation, and recognized tasks, which consisted of a pick-and-follow task and a push-and-trace task. Points on the curving parameter line show the timing to separate the primitive motions. For example, *Move* and *Approach* in the task are segmented by the magnitude of ${}^{\text{object}}\text{P}_{\text{finger}}$; *Grasp* is segmented by the gravitational direction element of ${}^{\text{ref}}\boldsymbol{F}_{\text{finger}}$; *Translate* is segmented by the gravitational direction element of ${}^{\text{ref}}\boldsymbol{p}_{\text{object}}$; *Follow* and *Release* are segmented by the norm of ${}^{ref}\boldsymbol{F}_{finger}$ and the contact flag.

After the segmentation of primitive motions, a motion sequence from *Move* to *Release* was grouped into a task, and a task type recognized. This showed that the segmentation of motion data and recognition of task were executed appropriately. After the recognition of task type, the primitive motions were relabeled based on the task understanding. For example, *Slide* between *Approach* and *Grasp* in the first pick-and-follow task was combined as *Grasp* because the *Slide* was a fluctuation by the operator. Similarly, the second *Push* between *Trace* and *Trace* in the second push-and-trace task was combined into *Trace*. After relabeling, robot finger trajectories were generated smoothly.

Desired position and force profiles were smoothed in order to reduce the vibration in robot hand motion. After performing an action, the position and orientation of the robot hand was determined to maximize the hand manipulability measure and robot commands were generated. The computer simulation was then executed to check the robot commands, as shown in Figure 6(a), in which the CG of primitive motions are presented. Finally, the tasks were executed experimentally by the 6-DOF robot arm with the Gifu Hand III, as shown in Figure 6(b). These images show that the proposed segmentation can be applied to robot teaching that includes performance of plural tasks.

http://dx.doi.org/10.5772/59189

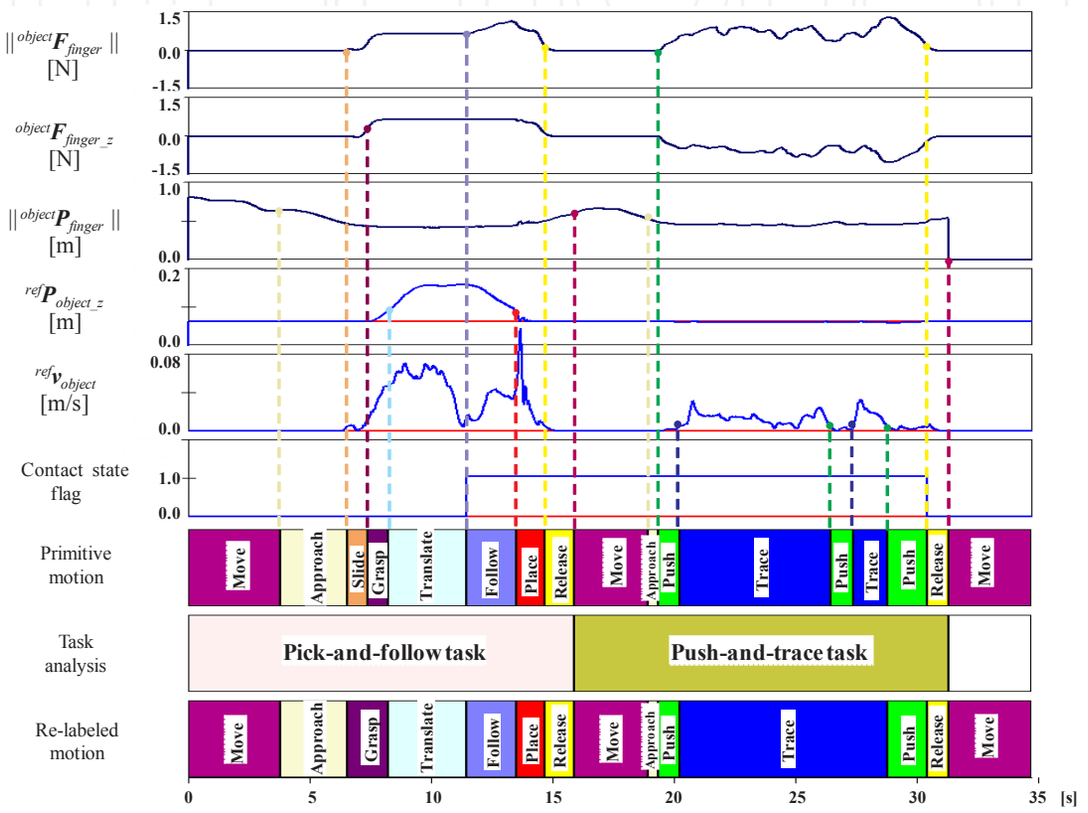
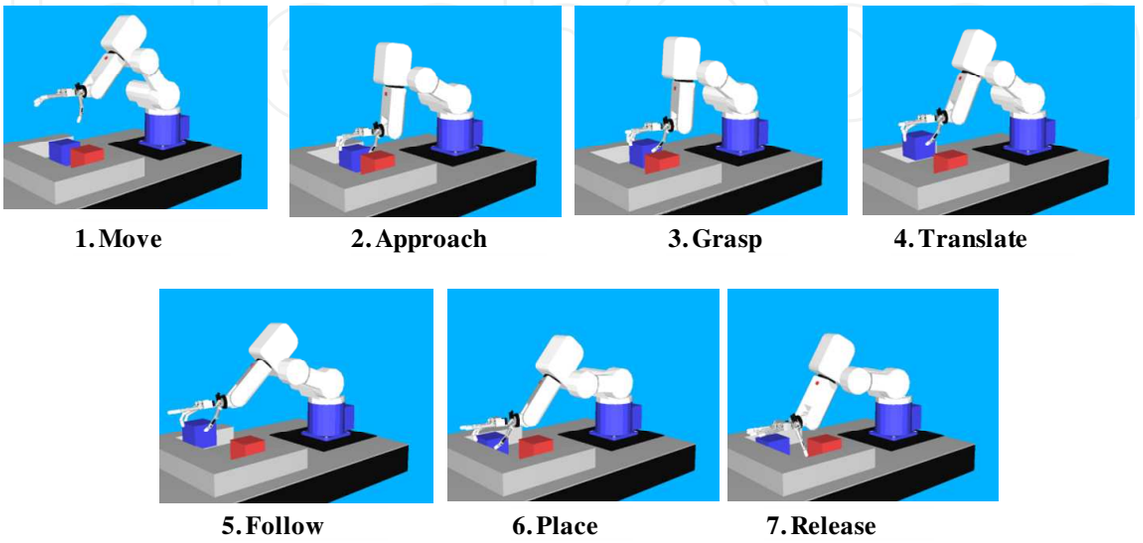
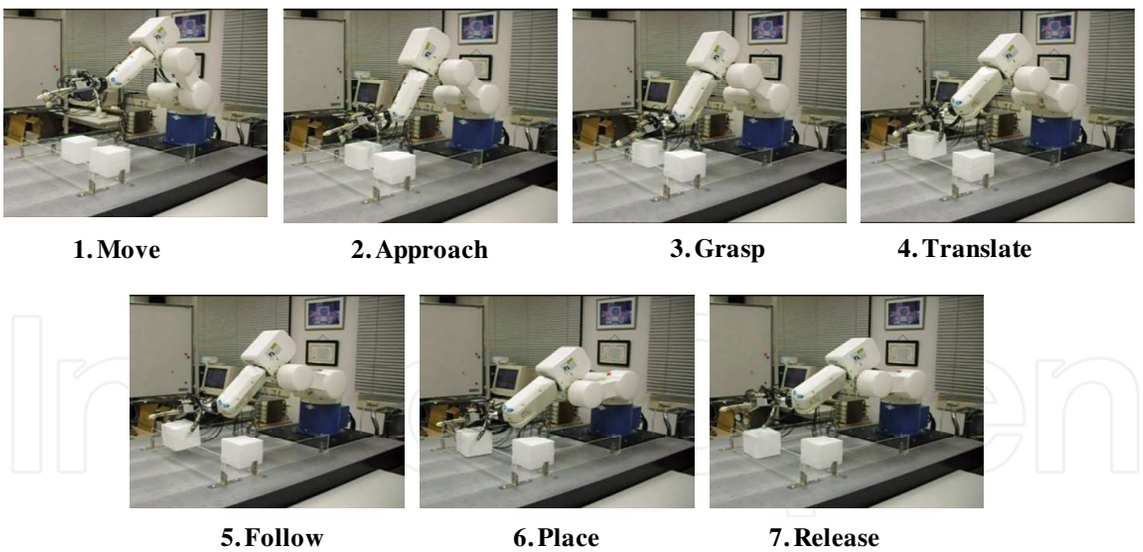


Figure 5. Segmentation of motion data and task analysis

The force profile at the thumb fingertip is shown in Figure 7. The x, y, and z elements of the force show the friction depending on contact force between the object and the wall, the normal contact force between the object and fingertip of the thumb, and the friction caused by gravitational force, respectively. The total force acting on the object is shown in Figure 8. These show the contact timings between finger and object, and object and wall. Experimental results almost follow the desired profiles. The robot could execute the task, and the operator could feel a realistic 3D force.



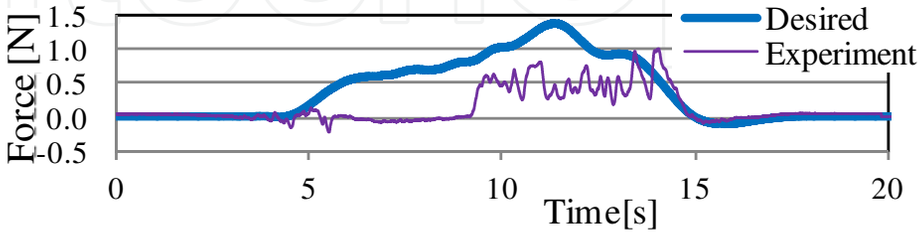
(a) Robot simulation



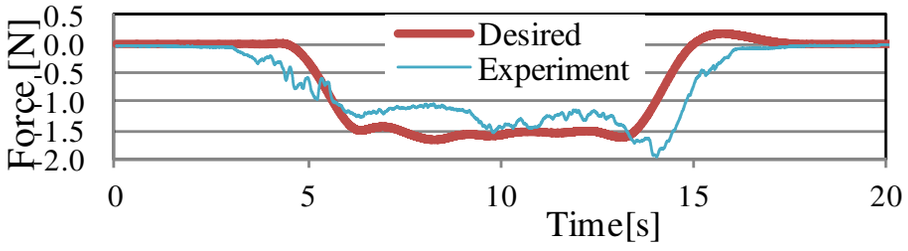
(b) Robot experiment

Figure 6. Robot simulation and experiment of the pick-and-follow

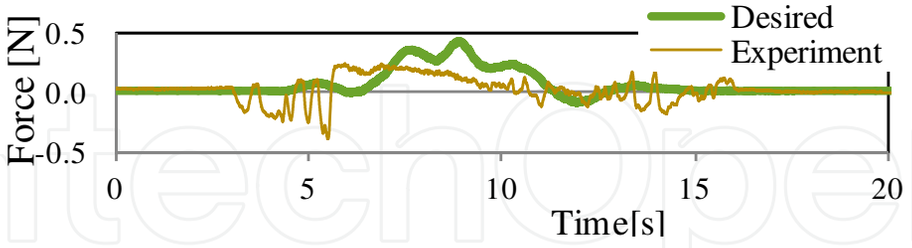
http://dx.doi.org/10.5772/59189



(a) x axis

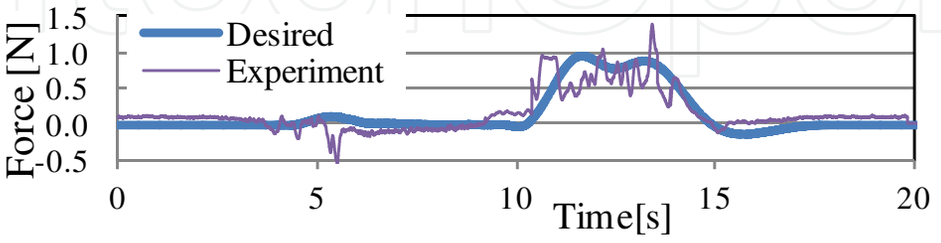


(b) y axis

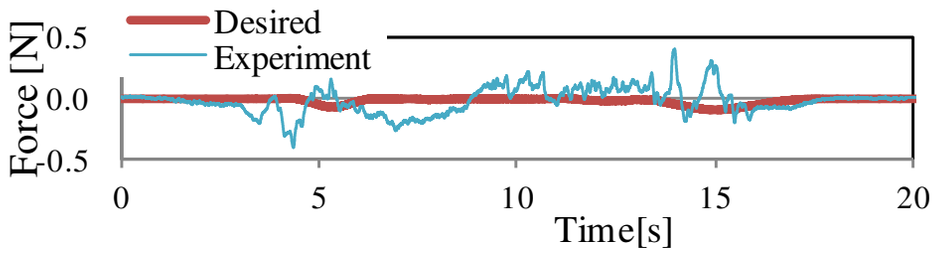


(c) z axis

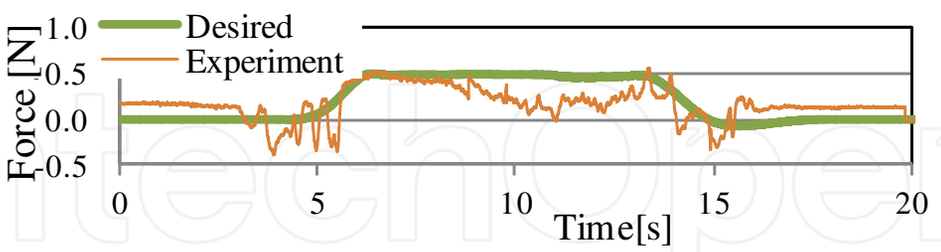
Figure 7. Thumb fingertip force



(a) x axis



(b) y axis



(c) z axis

Figure 8. Total force acting on the object

http://dx.doi.org/10.5772/59189

4. Virtual robot teaching for both-hand robots

Both-hands robots are expected to execute more varied and complicated tasks than single-hand robots, because both-hands robots can accomplish single-hand tasks with each hand independently, as well as both-hands coordinated tasks. We extended the VR robot teaching for single-hand robots in Section II to VR robot teaching for both-hands robots. The basic approach was to distinguish between single-hand task and both-hands coordinated tasks.

4.1. Work environment

The experimental system is shown in Figure 9. An operator manipulates virtual objects in CG through the bimanual multi-fingered haptic interface robot HIRO II. A peg-in-hole task in CG is shown in Figure 10 as an example of a work environment. The ten white spheres show the fingertip positions of the human measured by the HIRO II. The virtual objects are a circular ring and a cylinder with a pole. The VR system checks physical interference between the following geometrical relations: fingertip and the virtual object, the virtual object and environment, and one virtual object and the other. The position and orientation of virtual objects are transformed based on physical dynamics models.

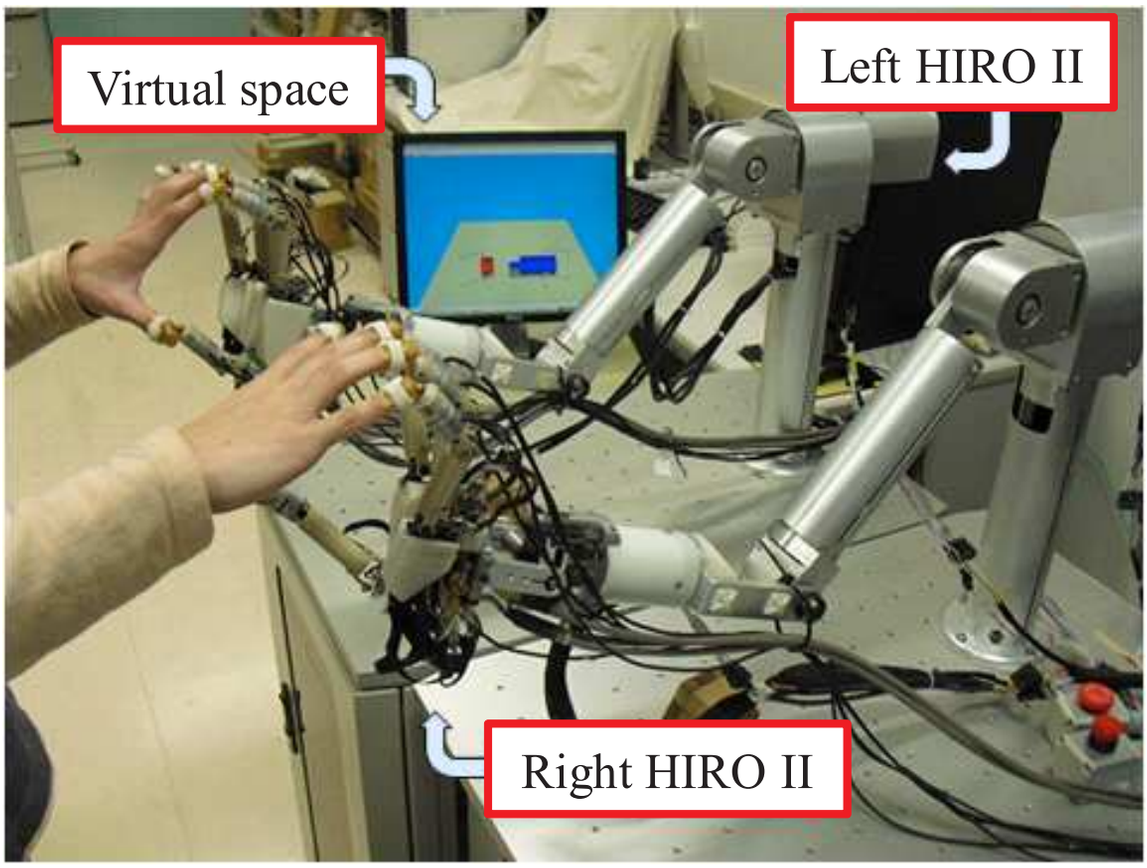


Figure 9. Bimanual multi-fingered haptic interface

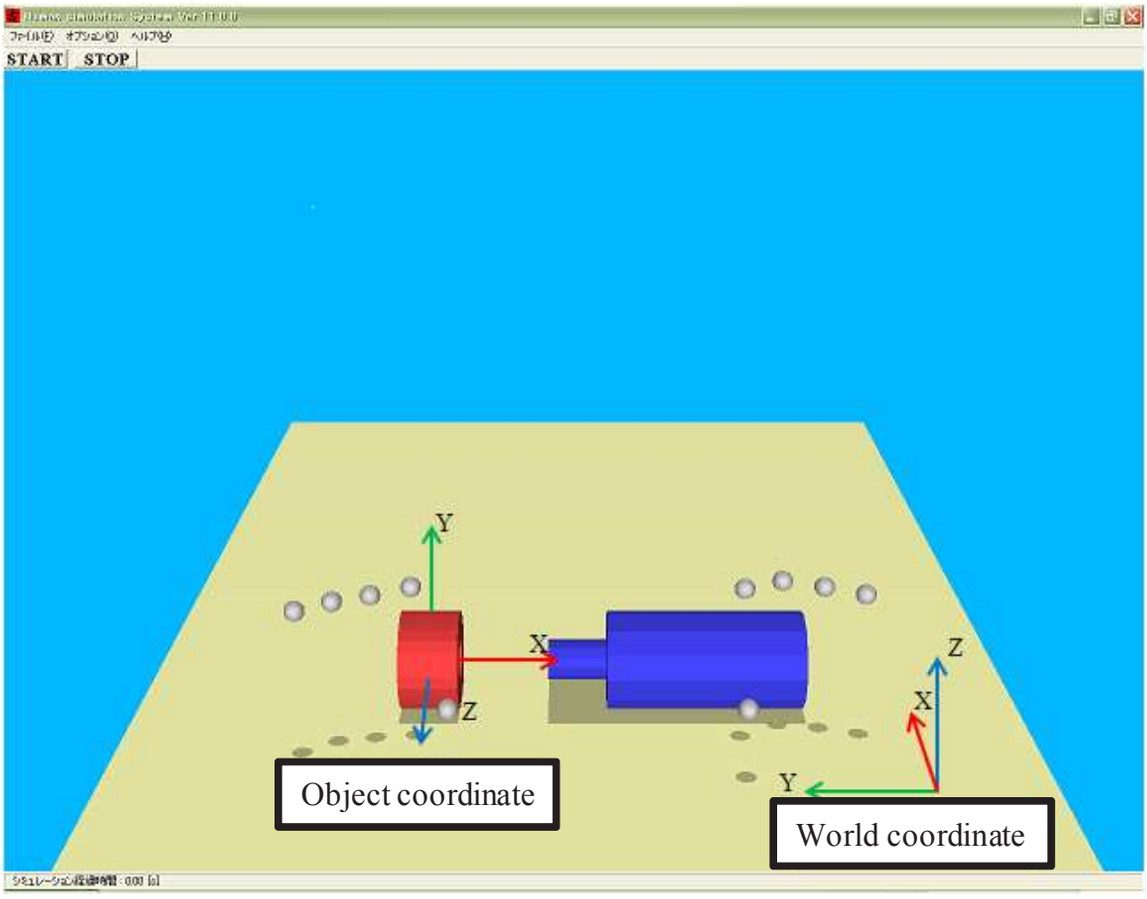


Figure 10. Virtual task environment

4.2. Segmentation and task analysis

The conceptual scheme of the VR robot teaching system was described in Section II. In this section, the system is extended to both-hands task. Both-hands task means bimanual coordination work, such as *Push-and-trace*, *Pick-and-follow*, *Peg-in-hole*, and *Pick-and-place,* etc. For example, the case in which an object is grasped in each hand and the two objects are brought into contact in air by the left and right hands would be considered a *Both-hand task*. Figure 11 is a decision flowchart for *Both-hand task*. In the figure, both *Left-hand task* and *Right-hand task* are related to the flow chart in Fig. 3. This part is an appended segmentation tree for both-hands tasks.

When humans translate an object using both hands, the sequence of motion of the work can be divided into plural primitive motions based on grasping state, work flow, and so on. These primitive motions include human intention. The robot should be controlled based on the human intention. We assume that both-hand tasks consist of the following 5 additional primitive motions: *Equal Grasp, Main Grasp, Pass, Fit,* and *Translate*. In the *Equal Grasp* segment, both hands grasp an object and translate it with equal contact force. In *Main Grasp,* the left or right hand has the main grasp on an object, and the other hand's grasp is ancillary. In *Pass,* one hand grasps an object and passes it to the other hand. The object is then translated to the

http://dx.doi.org/10.5772/59189

target position. In *Fit* the left and right hands translate two objects individually, then bring the two into contact. In *Translate*, the two objects are translated by the left and right hands individually, and do not interfere with each other. The segmentation tree for these primitive motions is shown in Figure 12. When new primitive motion is needed, the segmentation tree is additive by simple modification.

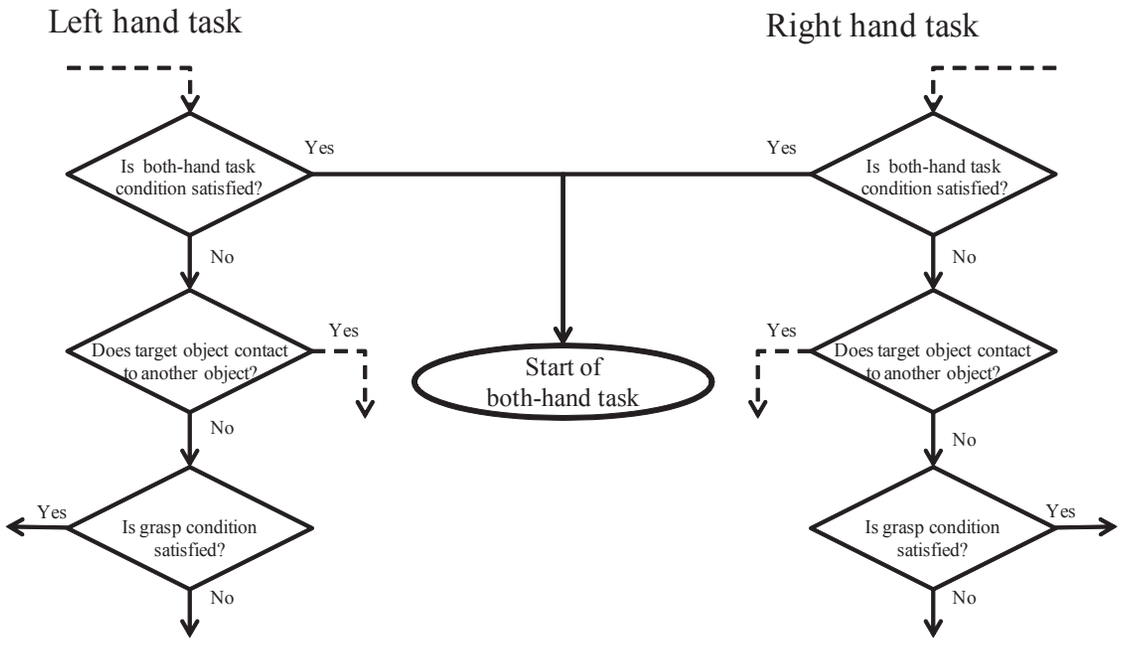


Figure 11. Flow chart of checking bimanual task

Task type is analyzed based on the sequence of the obtained primitive motions in the manner described in Section II-E. A sequence of primitive motions from *Move* to *Release* is considered to be one task, because the hand must move to the target object to accomplish something first, and must release the object at the end of the task. In task recognition, the task is recognized by a key primitive motion. For example, if *Insert* is in the sequence of primitive tasks, the task is a *Peg-in-hole task*; if the sequence of primitive tasks involves *Push*, the task is a *Push-and-trace task*.

After task recognition, primitive motions are relabeled based on the recognized task. For example, *Fit* between *Translate* and *Translate* is combined as *Translate* because *Fit* is a fluctuation by the operator. This modification process will be added as appropriate when a new primitive motion is added. After the relabeling based on the recognized task, the desired trajectories, contact points, and contact forces are analyzed within each segment.

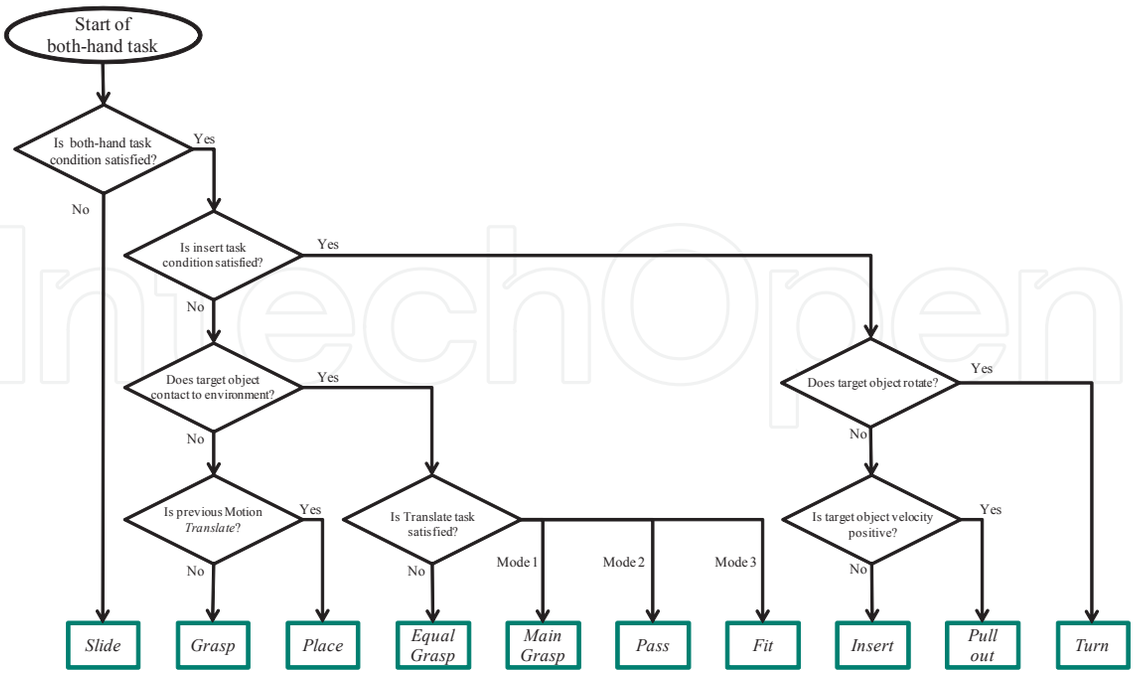


Figure 12. Flow chart of bimanual tasks

4.3. Experiment of Pick-and-Place Task

We executed VR robot teaching in a virtual environment through bimanual multi-fingered haptic interfaces, as shown in Figure 13. The operator executed a pick-and-place task, in which he moved his left hand to the target object, grasped and picked it up with the left hand, passed it to right hand, grasped it with the right hand, translated it to a target position, and released it. Figure 14 shows the measured parameters, primitive motions obtained by the proposed segmentation, and the recognized task, which consisted of two pick-and-place tasks. Points on the curving parameter line show the timing to separate the primitive motions as explained in Section II-B. For example, in bimanual operation, *Main Grasp* and *Equal Grasp* are segmented by the norm of ${}^{ref}\boldsymbol{F}_{finger}$ and the contact flag.

After the segmentation of primitive motions, a motion sequence from *Move* to *Release* was grouped into a task, and a task type recognized. This showed that the segmentation of motion data and recognition of task were executed appropriately. Once the task type was recognized, the primitive motions were relabeled based on the task understanding. For example, in bimanual operation, the sequence *Main Grasp* by left hand, *Equal Grasp*, and *Main Grasp* by right hand were combined into *Pass*. After relabeling, robot finger trajectories were generated smoothly.

Desired position and force profiles were smoothed to reduce the vibration in robot hand motion. After performing a task, the position and orientation of the robot hand was decided to maximize the hand manipulability measure [21] and robot commands were generated. The computer simulation was then executed to check the robot commands, as shown in Figure 15, in which the CG of primitive motions is presented.

http://dx.doi.org/10.5772/59189

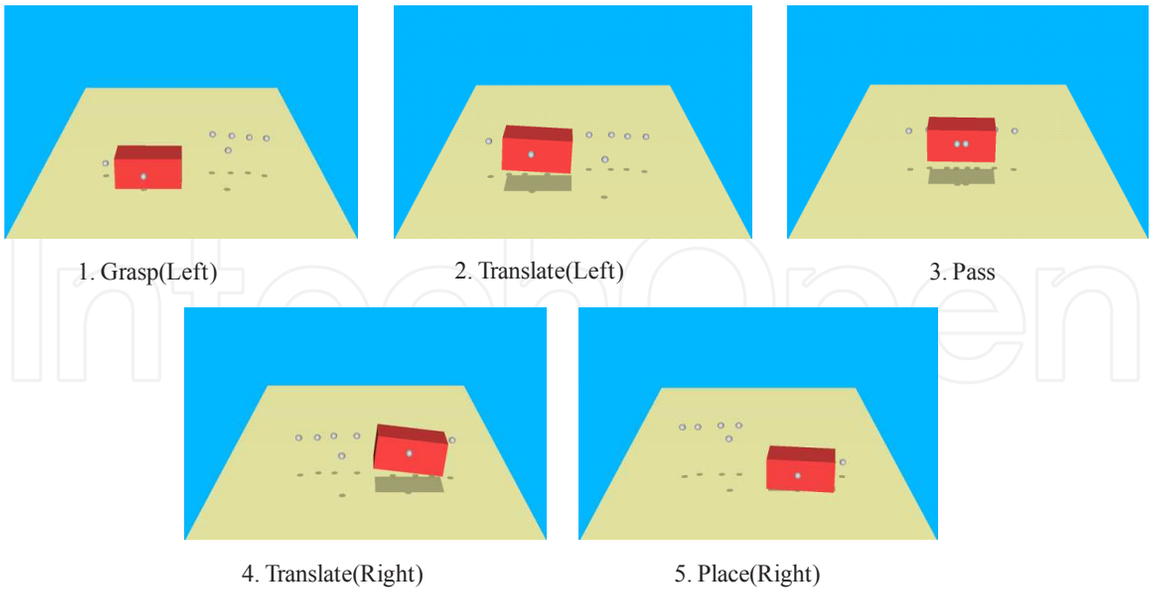


Figure 13. Human simulation

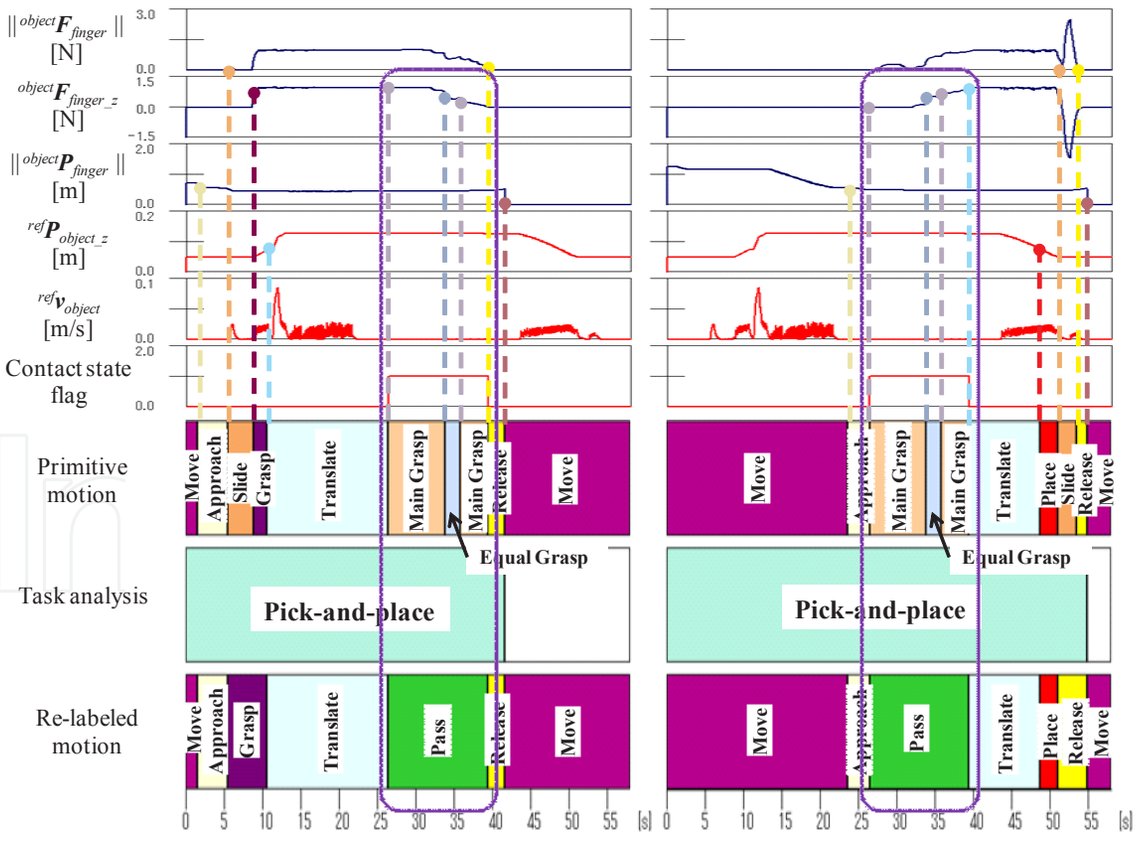


Figure 14. Segmentation of motion data and task analysis

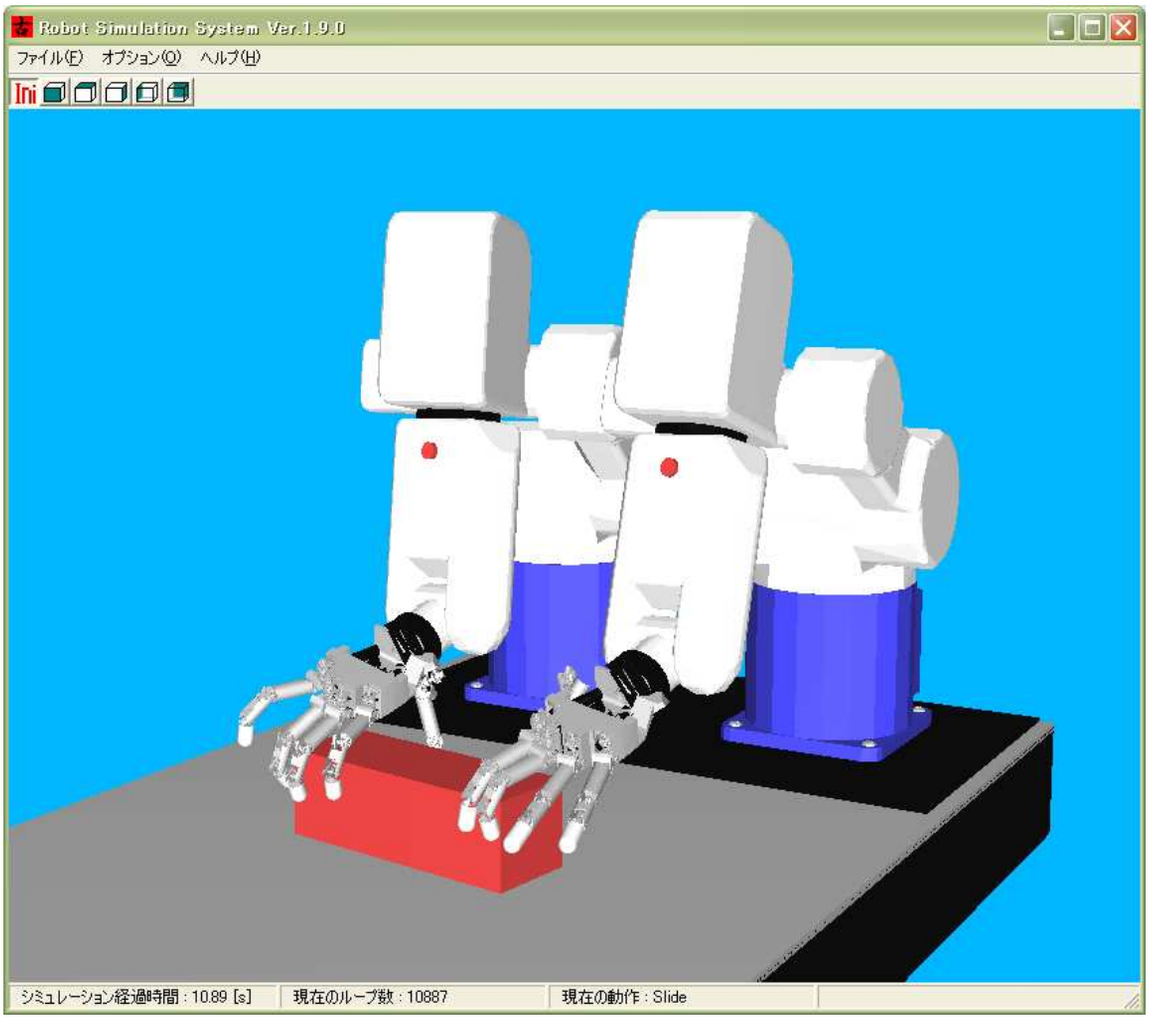


Figure 15. Robot simulation

5. Conclusions

We presented a VR robot teaching system, consisting of human demonstration and motion-intention analysis in a virtual reality environment using a multi-fingered haptic interface for automatic programming of multi-fingered robots. This approach has extended VR robot teaching to bimanual tasks using both-hand multi-fingered haptic interfaces. By using 3D forces at contact points between human fingers and an object, new tasks, including contact with multiple objects, can be learned in a virtual reality environment. The segmentation is executed according to the proposed segmentation tree, which is additive for new primitive motions. Task type is analyzed based on the obtained sequence of primitive motions, and the primitive motions are relabeled based on the recognized task. This method permits us to demonstrate plural tasks sequentially in a virtual reality environment.

http://dx.doi.org/10.5772/59189

This approach makes the virtual teaching system user-friendly. Our experimental results for performing an assembly task using a humanoid robot hand named Gifu Hand III and a multi-fingered haptic interface robot HIRO II demonstrate the effectiveness of the proposed method. Furthermore, we demonstrated that the VR robot teaching method can be extended to both-hands robot teaching.

Acknowledgements

This paper was supported in part by SCOPE (No. 121806001), by the Ministry of Internal Affairs and Communications, and by a Grant-in-Aid for Scientific Research from JSPS, Japan ((A) No. 26249063). The authors would like to thank the members of our laboratory, and in particular, Mr. Syunsuke Nanmo for his cooperation with the experiments.

Author details

Haruhisa Kawasaki[1*], Tetsuya Mouri[1] and Satoshi Ueki[2]

*Address all correspondence to: h_kawasa@gifu-u.ac.jp

1 Department of Mechanical Engineering, Gifu University, Gifu, Japan

2 Department of Mechanical Engineering, Toyota National Colleges of Technology, Toyota, Japan

References

[1] H. Asada, Y. Asari The Direct Teaching of Tool Manipulation Skills via the Impedance Identification of Human Motion, In Proceedings IEEE International Conference Robotics and Automation, pp. 1269-1274, 1988.

[2] T. Kuniyoshi, M. Inaba, H. Inoue, Teaching by Showing: Generating Robot Programs by Visual Observation of Human Performance, In Proceedings 20th International Symposium Industrial Rrobots, pp. 119-126, 1989.

[3] K. Ikeuchi, T. Suehiro, Toward an Assembly Plan from Observation – Part 1: Task Recognition with Polyhedral Objects, IEEE Transactions on Robotics and Automation, Vol. 10, No. 3, pp. 368-385, 1994.

[4] S.B. Kang, K. Ikeuchi, Toward Automatic Robot Instruction from Perception-Temporal Segmentation of Tasks from Human Hand Motion, IEEE Transactions on Robotics and Automation, Vol. 11, No. 5, pp. 670-681, 1995.

[5] R. M. Voyles, J. D. Morrow, P. K. Khosla, Gesture-Based Programming for Robotics: Human-Augumented Software Adaptation, IEEE Intelligent Systems, November/ December, pp. 22-29, 1999.

[6] J. Kulick, M. Toussaint, T. Lang and M. Lopes, Active Learning for Teaching a Robot Grounded Relational Symbols, Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, pp. 1451-1457, 2013

[7] S. Calinon and A. Billard, Active Teaching in Robot Programming by Demonstration, 16th IEEE International Conference on Robot & Human Interactive Communication, pp. 702-707, 2007.

[8] L. Peternel and J. Babiˇc, Humanoid Robot Posture-Control Learning in Real-Time Based on Human Sensorimotor Learning Ability, IEEE International Conference on Robotics and Automation (ICRA), pp. 5329-5334, 2013

[9] H. Kitagawa, T. Terai, P. Minyong, and K. Terashima, Application of Neural Network to Teaching of Massage using Muti-fingered Robot Hand, Journal of Robotics and Mechatronics, Vol. 14, no.2, pp.162-169, 2002

[10] L. Rozo, P. Jiménez and C. Torras, Learning Force-Based Robot Skills from Haptic Demonstration, Proceedings of International Conference on Applied Bionics and Biomechanics, 2010

[11] J.-G. Ge, Programming by Demonstration by Optical Tracking System for Dual Arm Robot, Proc. of 44th International Symposium On Robotics (ISR2013), pp. 1-7, 2013

[12] S. Lee, M. Kim band C.-W. Lee, Human-robot integrated teleoperation, Advanced Robotics, Vol. 13, No. 4, pp. 437-449, 1999

[13] H. Lee1 aand J. Kim, A survey on Robot Teaching: Categorization and Brief Review, Applied Mechanics and Materials Vol. 330 (2013) pp 648-656, Trans Tech Publications, pp.648-656, 2013

[14] M. Kaiser, R. Dillmann, Building Elementary Robot Skills from Human Demonstration, in Proceedings IEEE International Conference Robotics and Automation, pp. 2700-2705, 1996.

[15] T. Sato, Y. Nishida, J. Ichikawa, Y. Hatamura, H. Mizoguchi, Active Understanding of Human Intention by a Robot through Monitoring of Human Behavior, Transactions of JRSJ, No. 4, Vol. 13, pp. 545-552, 1995 (in Japanese).

[16] M. Tsuda, H. Ogata, Y. Nanjo, Programming Groups of Local Models from Human Demonstration to Create a Model for Robotic Assembly, in Proceedings ICRA, pp. 530-537, 1998.

[17] H. Onda, T. Suehiro and K. Kitagaki, Teaching by Demonstration of Assembly Motion in VR-Non-deterministic Search-type Motion in the Teaching Stage-, Proceed-

http://dx.doi.org/10.5772/59189

ings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3066-3072, 2002

[18] H. Kawasaki, K. Nakayama, G. Parker, Robot Teaching Based on Intention of Human Motion In Virtual Reality Environment, TVRSJ Vol. 5, No. 2, pp. 899-906, 2000. (in Japanese).

[19] S. B. Kang, K. Ikeuchi, Toward Automatic Robot Instruction from Perception-Mapping from Human Grasps to Manipulator Grasps, IEEE Transactions on Robotics and Automation, Vol. 13, No. 1, 1997.

[20] R. N. Rohling, J. M. Hollerbach, Optimized Fingertip Mapping for Teleoperation of Dexterous Robot Hand, in Proceedings IEEE International Conference on Robotics and Automation, pp. 769-775, 1993.

[21] H. Kawasaki, K. Nakayama, T. Mouri, S. Ito, Virtual Teaching Based on Hand Manipulability for Multi-Fingered Robots, In Proceedings of IEEE International Conference on Robotics and Automation, pp. 1388-1393, 2001.

[22] H. Kawasaki, T. Furukawa, S. Ueki, T. Mouri, Virtual Robot Teaching Based on Motion Analysis and Hand Manipulability for Multi-Fingered Robot, Journal of Advanced Mechanical Design, Systems, and Manufacturing, Vol. 3, No. 1, pp. 1-12, 2009.

[23] H. Kawasaki, S. Nanmo, T. Mouri, S. Ueki, Virtual Robot Teaching for Humanoid Hand Robot Using Muti-Fingered Haptic Interface, Proceedings of 2011 IEEE International Conference on Communications, Computing and Control Applications (CCCA'11), pp. 1-6, 2011

[24] T. Mouri, H. Kawasaki, K. Yoshikawa, J. Takai, S. Ito, Anthropomorphic Robot Hand: Gifu Hand III, In Proceedings of International Conference ICCAS2002, pp. 1288-1293, 2002.

[25] H. Kawasaki, T. Mouri, Design and Control of Five-Fingered Haptic Interface Opposite to Human Hand, IEEE Transaction on Robotics, Vol. 23, No.5, pp. 909-918, 2007.

[26] M. R. Cutkosky, On Grasp Choice, Grasp Models, and the Design of Hands for Manufacturing Tasks, IEEE Transactions on Robotics and Automation, vol. 5, No.3, pp. 269-279, 1989.

IntechOpen

IntechOpen