Human Robot Cooperation for Mechanical Assembly using Cooperative Vision System

H.Kimura and T.Horiuchi
Graduate School of Information Systems, Univ. of Electro-Communications
1-5-1 Chofu-ga-oka, Chofu, Tokyo 182

K.Ikeuchi
Institute of Industrial Science, Univ. of Tokyo
7-22-1 Roppongi, Minato-ku, Tokyo 106

Abstract

In order to assist the human, a robot must autonomously recognize the human motion in real time. The vision is the most useful sensor for this purpose. The robot recognizes the current target objects and the human grasp by vision, and must plan and execute the needed assistance motion based on the task purpose and the context. In this research, we tried to solve such problems. We defined the abstract task model, analyzed the human demonstration by using events and a state buffer, and automatically generated the task models needed in the assistance by the robot. The robot planned and executed the assistance motion based on the task models in the cooperation with the human by analyzing the human motion. We implemented the 3D object recognition system and the human grasp recognition system by using the trinocular stereo color cameras and the real time range finder. The effectiveness of these methods was tested through an experiment in which the human and the robotic hand assembled toy parts in cooperation.

1 Introduction

For the purpose of the child care and nursing care, we are developing the robot which can assist the human in a cooperative task. In order to assist the human, a robot must autonomously recognize the human motion in real time. The vision is the most useful sensor for this purpose. The robot recognizes the current target objects and the human grasp by vision, and must plan and execute the needed assistance motion based on the task purpose and the context.

In this study, we are constructing the human robot cooperation system in which a robot can recognize the motion of the human in real time without any special facilities such as a 'data glove' and can assist the human autonomously. As an example of a cooperative task, we take up the assembly of toy parts by the human and a robot hand (Fig.1). The assistance operations by a robot are generated based on a task model. Task models are created by observation of the human demonstration.

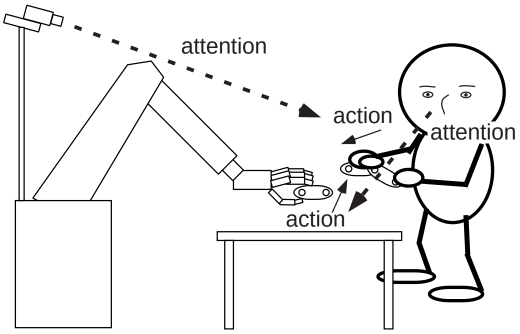


Figure 1: Cooperation between human and robot. Two vision systems communicate via actions.

In order to analyze the human motion in demonstration and in cooperation, we implemented the 3D object recognition system and the human grasp recognition system by using the trinocular stereo color cameras and the real time range finder. In order to create a task model from the human demonstration and to plan the robot operations in cooperation, we defined the abstract task model and implemented events and a state buffer mechanism.

We can tell that observation of the human motion by vision and cooperation control based on the task models are the interesting application of the "Cooperative Distributed Vision" to the robotics field when we consider that two vision systems such as the human and a robot share the physical world, and communicate and cooperate via actions (Fig.1).

2 Robot System

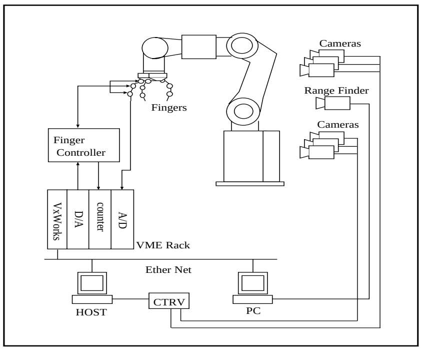


Figure 2: Robot system

The robot system is shown in Fig.2. The robot hand is mounted on the 7-dof arm. The robot hand with three fingers manufactured is controlled by VxWorks. Each finger has three joints and 6-axes force/torque sensor. The trinocular stereo color cameras connected to the color tracking vision board. and the real time range finder are used for the vision system. The real time range finder can generate 24x24 depth image in video rate.

3 Visual Processing

3.1 Recognition of Target Object

The 3D object recognition using the appearance model and minute templates [9] has advantages such as unnecessity of the extraction of the geometrical features, unnecessity of the segmentation at the first stage and robustness for occlusion. In this paper, we implemented 3D object recognition method on a 2D image (Fig.3, 4) referring to [9] and 3D position measurement of minute templates by subpixel stereo.

3.2 Recognition of Human Grasp

3.2.1 Assembly Task and Human Grasp

Cutkosky [10] pointed out that the human used various grasps of tools according to a task and the shape of a tool, and analyzed the human grasp in view of dexterity and precision. The ordinary grasp of a driver, wrench and toy parts are shown in Fig.5. But, another grasp for the same tool may be used when the purpose of the task is different. For an example, the power grasp different from Fig.5-(a) is used in order to fix a screw hardly. This means that there exists the physical relation between the human intention for the task and the human grasp. For this reason, we can tell that observing how the human grasps a tool is important in order to determine the function in the grasped object to which the attention is being paid and to plan the assistance operations to the function.

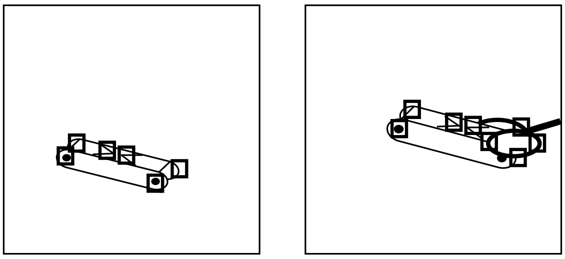


Figure 3: Minute templates on data image(left) and current scene(right) generated by considering trackability

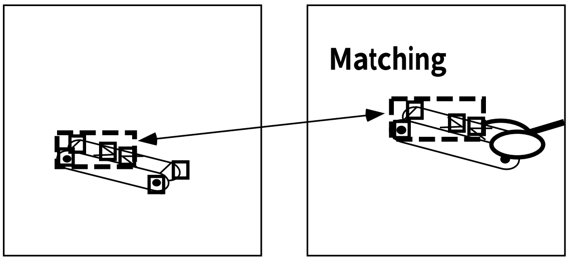


Figure 4: Templates cluster matched between data image(left) and current scene(right) by considering correlation and geometric constraints

In addition, for the planning of the motion of the robot with an arm and fingers, the task model must involve the grasp information such as the position and orientation of the hand relative to the grasped object, configuration of fingers, the contacting points between an object and fingers, and so on. Therefore, the recognition of the human grasp is essential for automatic generation of task models for the robot hand with fingers [5].

Although the recognition of the human hand configuration or the human gesture [12] is actively studied

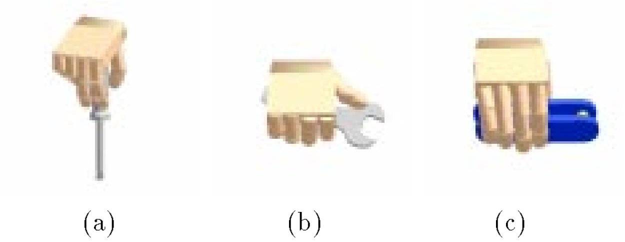


Figure 5: Human hand configuration models with a tool

on in the field of the human interface in these days, the above mentioned grasp information is important in the case of cooperation for mechanical assembly.

3.2.2 Human Grasp Recognition with Color Images and Depth Images

For the robot assistance for mechanical assembly, the online recognition of the human grasp is needed. In our vision system, the trinocular stereo cameras is not suitable for the real time recognition of the human hand configuration and the real time range finder cannot cover the wide area. Therefore, we combine these two sensors for the online recognition of the human grasp. The trinocular stereo cameras and the real time range finder observe the human hand from the almost same view.

For the recognition of the human hand configuration, 16x16 depth data images of the human hand configuration (Fig.6) were taken and the eigen-space [11] was created from these depth data images. The process of recognizing the human hand configuration and detecting the finger tips are as follows:

(1) The human hand area is searched on the current scene image taken by the center color camera by using color labeling, which is executed on the color tracking vision(CTRV) board using its hardware function.

(2) When the human hand area gets into the area covered by the range finder(RF), a 24x24 depth image is taken by RF and the clipped area on the depth image corresponding to the labeled hand area is normalized to the 16x16 depth image (Fig.7-A). In order to eliminate the pixels for a grasped object or background, the pixels not corresponding to the human hand are cleared to zero based on the color of the corresponding pixel of the current color image before normalization,

(3) The human hand configuration is recognized by projecting this 16x16 depth image into the data images eigen-space (Fig.7-B). As a result, the depth data image matched with the current hand configuration are obtained (Fig.7-C).

(4) The areas of finger tips registered on the color data image taken at the same time with the depth data image (Fig.7-C) are searched on the current scene image by using CTRV hardware function (Fig.7-D).

(5) The 3D positions of the areas of finger tips found in (4) are measured by trinocular stereo cameras by using CTRV hardware function.

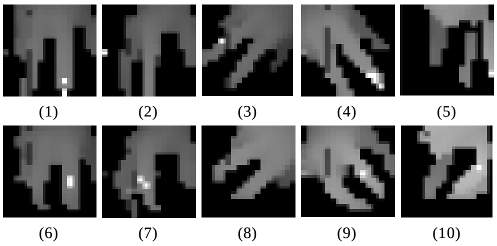


Figure 6: Examples of depth data images

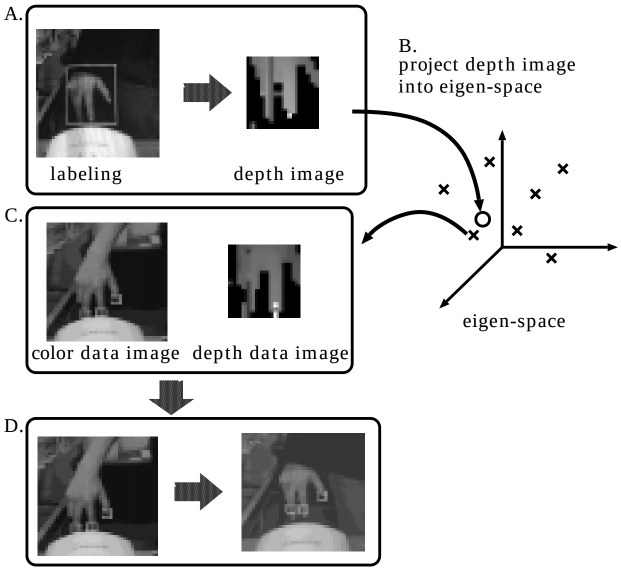


Figure 7: Recognition process of the human hand configuration and positions of finger tips

3.3 Experiments

The results of experiments are shown in Fig.8. The processes (1)-(5) in **3.2.2.** were executed in every

0.6 (sec.). The labeled areas of the human hand, the 16x16 depth images of the human hand and the found areas of finger tips are shown in Fig.8-(a), (b) and (c), respectively. The upper image and lower image of Fig.8-(b) are matched with the depth image of Fig.6-(1) and Fig.6-(8), respectively.

The successful results of recognition experiments of the human hand configuration with tools are shown in Fig.9. But, since the recognition of the human grasp by matching the human hand configuration with the human hand model is not sufficient at present, the automatic detection of grasping points and the mapping of the human grasping to the robot hand are left as the future works.

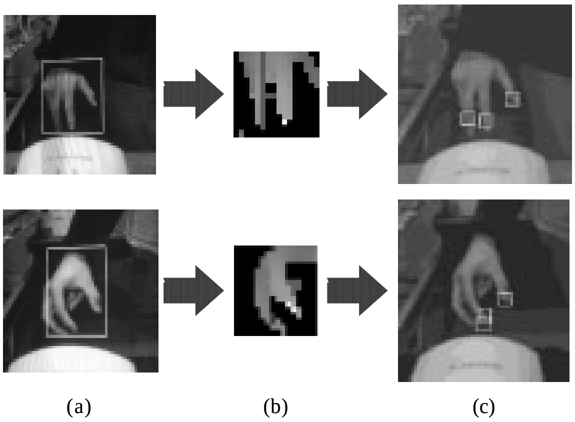


Figure 8: Results of finger tip recognition

4 Analysis of Human Demonstration and Generation of Task Model

4.1 Definition of Task Model

In this paper, we assume the following attributes about target objects are given:

- the frame of a target object on the appearance model data image, which is calibrated by the stereo cameras,
- frames of the functions such as a hole, an axis, etc. in the frame of a target object.

We define the following notations about the frames of the functions of a object and operations by the human hand or a robot hand.

- 'obj(a).f(i)' means the i-th **function** of the object 'a'. In the case that the function is not specified, '.f(i)' is eliminated.
- 'operate obj(a).f(i) {to obj(b).f(j)}' means an **operation** to functions or a function. 'obj(a).f(i) {& obj(b).f(j)} operated' means a **state** of functions or a function caused by the operation. The part '{}' means that the operation affects two functions.

Under such definitions, the **abstract task model** consists of the following two elements (Fig.10):

- the state of a function or the state of fixed functions of two objects,
- the preconditions for the state of functions.

For examples, 'obj(a) grasped' is the state 'grasped' of the unspecified function of the object 'a'. 'obj(a).f(i) & obj(b).f(j) fixed' is the sate 'fixed' of paired functions such as the shaft-hole pair, the driver-screw pair and two faces attached, where the sate 'fixed' means that points and axes of two functions are coincided and aligned, respectively. The preconditions are also states of functions. For an example, the fixation of a hole and a shaft is the precondition of the screwing of the shaft by a driver.

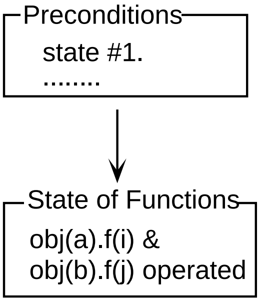


Figure 10: Abstract task model: obj(a).f(i) means the i-th function of object named #a.

4.2 Generation of Task Model

For the each scene of the human demonstration, the 3D recognition of objects (toy parts and tools) and the human hand configuration is executed by the visual processing described in **3.** and the relations between functions of objects and the human hand grasping are analyzed. The results of the 3D recognition and the analysis are shown in Fig.11. The geometric models of the recognized objects and the area of the hand grasping a object are shown on each image.

The **human demonstration analyzer** (Fig.12) is able to execute the following procedures about the states of the toy parts and tools.

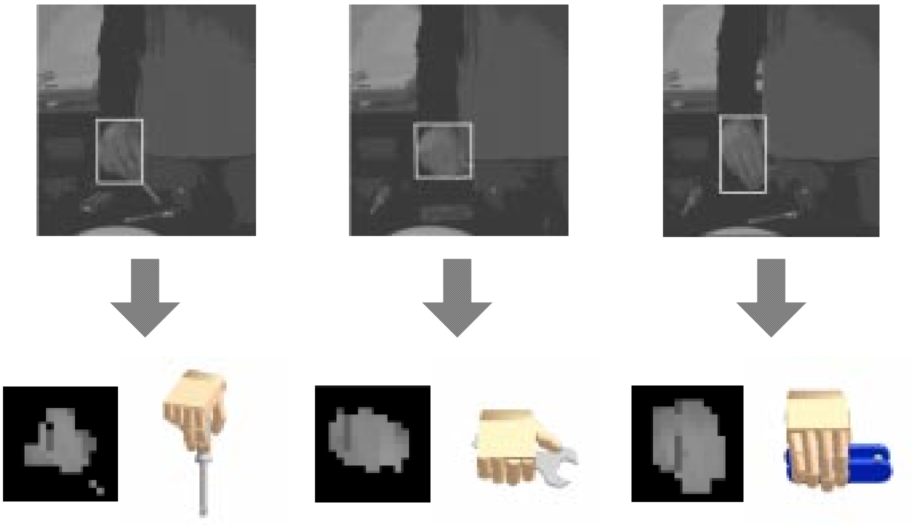


Figure 9: Recognition results of the human hand configuration with a tool

[a] The analyzer generates an **event** from the newly recognized state. An event has the same notation as a state.

[b] The analyzer recognizes the states such as 'obj(a) grasped' and 'obj(a).f(i) & obj(b).f(j) fixed'. The state 'fixed' is translated into the following states considering the coincided points and the aligned axes of the frames of functions.

obj(a).f(i) & obj(b).f(j) fixed
  obj(a).f(i).point & obj(b).f(j).point coincided
  obj(a).f(i).axis(k) & obj(b).f(j).axis(l) aligned

[c] The analyzer calculates parameters in order to generate the robot operation from the recognized state, and registers the parameters as attributes into the state (Fig.13). For an example, the axes which should be aligned in the operation 'fix obj(a).f(i) & obj(b).f(j)' are extracted from the state 'obj(a).f(i) & obj(b).f(j) fixed'.

[d] When the state 'obj(a).f(i) & obj(b).f(j) fixed' is recognized after 'obj(a).f(i) grasped' and 'obj(b).f(j) grasped', the analyzer translates these two states 'grasped' into one state 'obj(a-b) grasped' because the 'obj(a)' and 'obj(b)' are mechanically fixed.

By using the above described procedures, the human demonstration analyzer generates task models through the following process:

1) to push events detected at the present scene onto the stack called a **state buffer**,

2) to generate the task model when the state 'fixed' appears at the top of the state buffer,

3) to return to 1).

The events in the state buffer and task models generated by the analyzer for demonstration in Fig.11 are shown in Fig.14. At the process 2), the state 'fixed' on the top of the state buffer becomes the state of functions in the task model. Other states below the state 'fixed' on the state buffer become the preconditions in the task model. For examples, the task model #1 and #2 in Fig.15 are generated for events in the state buffer (b)-1 and (c) in Fig.14, respectively. The attributes in each state are calculated and registered into the state in a task model.

5 Recognition of Human Motion and Cooperative Assistance by Robot

5.1 Planning of Cooperative Assistance

The **cooperation planner** (Fig.12) analyzes the human operations by using events and the state buffer in the same way as the human demonstration analyzer. In addition, the cooperation planner divides the states in the task model obtained in **4.2.** into the states caused by the human operations and the states which should be caused by the robot operations. As a result, the cooperation planner can generate the robot operations for the assistance from the latter states.

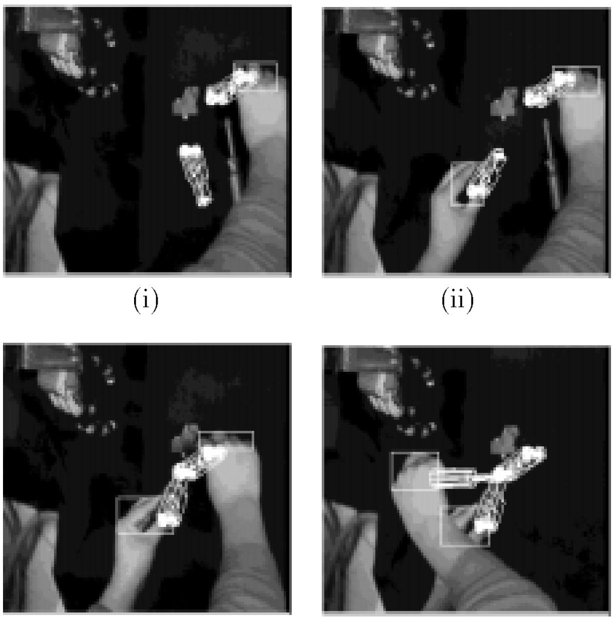

(i) (ii) (iii) (iv)

Figure 11: Example of analysis of human demonstration: parts axes fixation (iii) and screw hole-tip fixation (iv) are detected.

The cooperation planner is able to execute the following procedures in addition to the procedures [a]-[d] described in **4.2.**.

[e] The planner generates an operation from a state and its parameters registered as attributes (Fig.13). That is, the planner generates the trajectories of the arm and fingers.

[f] The planner confirms the event which should occur by the robot operation by using vision system and pushes the confirmed event on the state buffer.

By using the above described procedures, the assistance operations by the robot hand are generated from observation of human operations through the following process.

1) to push events detected at the present scene onto the state buffer,

2) to go to 3) when the part of preconditions of a task model registered in the system appears at the top of the state buffer, otherwise to return to 1),

3) to generate the robot operations from other preconditions and the state of functions in this task model,

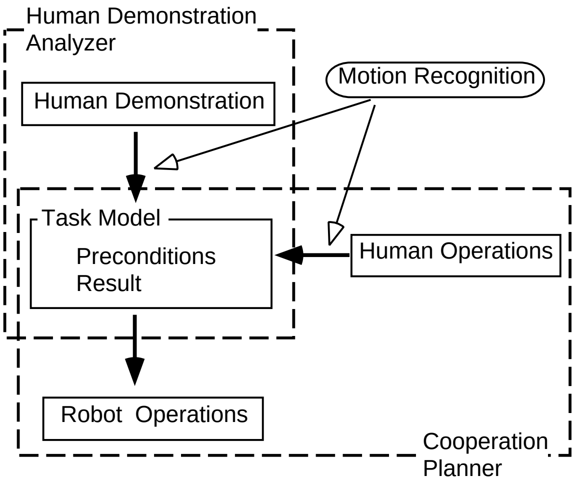


Figure 12: Overview of motion recognition based cooperation

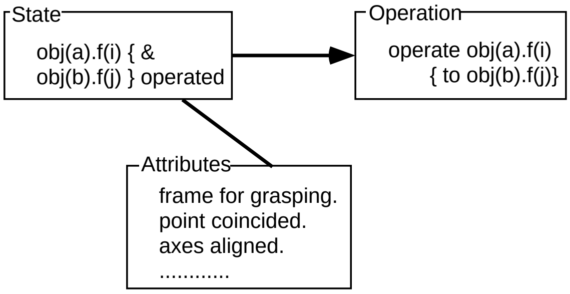


Figure 13: Operation is generated from state using its attributes

4) to execute these operations in turn if the robot hand is able to execute all operations, that is, to execute the assistance operations based on this task model,

5) to return to 1).

If all preconditions of a task model appears at the top of the state buffer at the process 2), no assistance operation based on this task model is needed.

5.2 Experiments

The scenes of cooperation and the process of generating the robot operations are shown in Fig.16- 17 and Fig.18- 19 for two experiments.

When the human had grasped #parts1 in Fig.16, the one of the preconditions, '#parts1 grasped', of

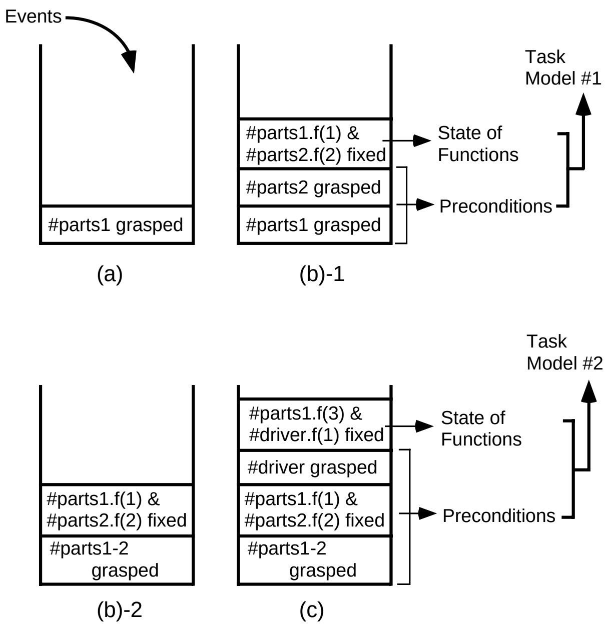


Figure 14: Events and state buffer: (a), (b)-1,2 and (c) are correspondent with (i), (iii) and (iv) of Fig.13, respectively.

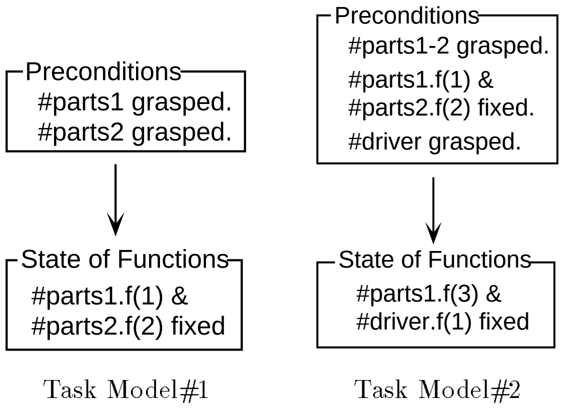


Figure 15: Task model generated by the analyzer for demonstration in Fig.13

the task model #1 appeared at the top of the state buffer in Fig.17. Therefore, the robot hand executed the assistance operations, 'grasp #parts2' and 'fix #parts2.f(2) to #parts1.f(1)' generated from another precondition '#parts2 grasped' and the state of functions '#parts1.f(1) & #parts2.f(2) fixed' in the task model #1, respectively (Fig.17), since they were possible to execute.

When the human had grasped #parts1 and #parts2 and fixed them in Fig.18, two of three preconditions, '#parts1-2 grasped' and '#parts1.f(1) & #parts2.f(2) fixed', of the task model #2 appeared at the top of the state buffer in Fig.19. Therefore, the robot hand executed the assistance operations, 'grasp #driver' and 'fix #driver to #parts1.f(3)' generated from another precondition '#driver grasped' and the state of functions '#parts1.f(3) & #driver.f(1) fixed' in the task model #2, respectively (Fig.19), since they were possible to execute.

6 Conclusion

In this study, we constructed the framework for the vision-based cooperation of the human and a robot hand for mechanical assembly. The task models for cooperation were generated by observation of the human demonstration. The operations of the robot assistance were generated based on the task models by observation of the human operations.

The following problems are left unsolved:

(1) The recognition of the human grasp is not sufficient for generating a task model of the robot hand.

(2) Since the task models obtained by observation of the human demonstration are rough models for many kinds of cooperations, the mechanism which makes models precise and adapt models to the present situation is necessary.

(3) The events generation mechanism at present is not sufficient for the continuous motion of the human.

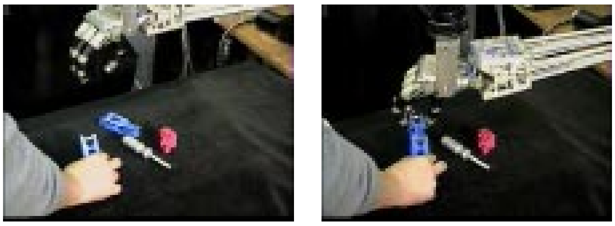
(4) The operation control mechanism using force sensors and skills is needed.

By solving such problems, the dynamic cooperation between the human and the robot via actions (Fig.1) becomes feasible.

References

[1] Y.Kuniyoshi, M.Inaba and H.Inoue, Learning by Watching: Extracting Reusable Task Knowledge from Visual Observation of Human Performance, IEEE Trans. on Robotics and Automation, Vol.10, No.6, pp.799-822, 1994.

[2] K.Ikeuchi, T.Suehiro, Towards an Assembly Plan from Observation Part I: Task Recognition With Polyhedral Objects, IEEE Trans. on RA, Vol.10, No.3, pp.368-385, 1994.



(a) (b)

Figure 16: Experiment1: parts axis fixation

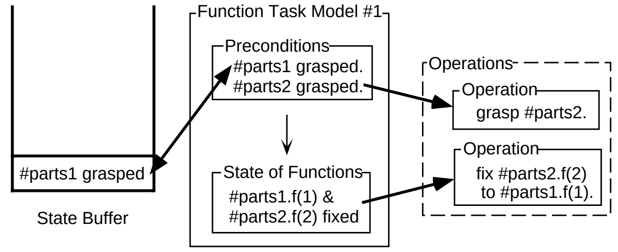


Figure 17: Operations of a robot hand are generated based on the task model #1

[3] Y.Xu, J.Yang and C.S.Chen, Gesture Interface: Modeling and Learning, Proc. of ICRA94, pp.1747-1752, 1994.

[4] R.M.Voyles and P.K.Khosla, Tactile Gestures for Human/Robot Interaction, Proc. of IROS95, pp.7-13, 1995.

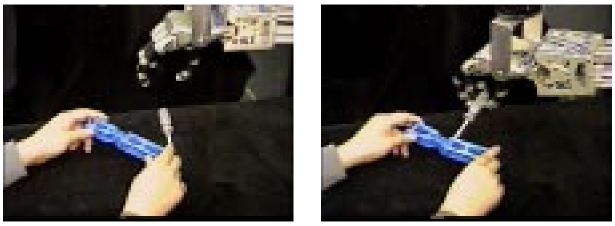
[5] S.B.Kang, K.Ikeuchi, Grasp Recognition and Manipulative Motion Characterization from Human Hand Motion Sequences, Proc. of ICRA94, pp.1759-1764, 1994

[6] Y.Kuniyoshi, Behavior Matching by Observation for Multi-Robot Cooperation, Proc. of ISRR'95, 1995

[7] H.Kimura and G.Kajiura, Motion Recognition Based Cooperation between Human Operating Robot and Autonomous Assistant Robot, Proc. of ICRA97, pp.297-302, 1997

[8] H.Kimura and H.Katano, Vision-Based Motion Recognition of the Hexapod for Autonomous Assistance, Proc. of IROS98, 1998

[9] K.Ohba and K.Ikeuchi, Recognition of the Multi Specularity Objects using the Eigen-Window, CMU-CS-96-105, 1996



(a) (b)

Figure 18: Experiment2: screw hole and driver tip fixation

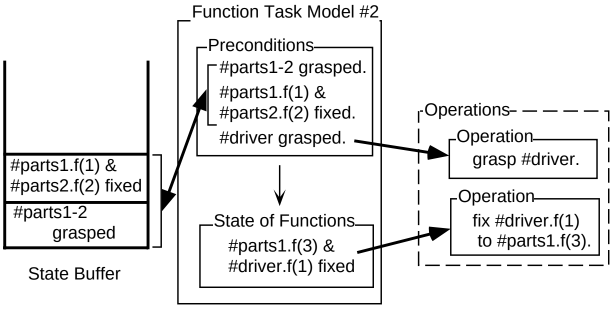


Figure 19: Operations of a robot hand are generated based on the task model #2

[10] M.R.Cutkosky and R.D.How, Human grasp choice and robotic grasp analysis, *Dextrous Robot Hands*, Springer-Verlag, pp.5-31, 1990

[11] H.Murase and S.K.Nayar, Visual Learning and Recognition of 3-D Objects from Appearance, Int. Journal of Computer Vision, Vol.14, No.1, pp.5-24, 1995.

[12] Y.Cui and J.Weng, Hand Segmentation Using Learning-Based Prediction and Verification for Hand Sign Recognition, Proc. of CVPR96, pp.88-93, 1996