ICAT '97

Direct Teaching to a Virtual Robot Arm

Shoujie He[1], Takao Horie[2], Kazuaki Tanaka[2],
Norihiro Abe[2], Jiangyu Zheng[2], and Hirokazu Taki[3]

[1]Dept of Computer Science and Electronics,
[2]Dept of Mechanical Systems Engineering,
Faculty of Computer Science and Systems Engineering,
Kyushu Institute of Technology
Kawazu 680-4, Iizuka, Fukuoka 820-8502, JAPAN
[3]Mitsubishi Electric Inc. JAPAN
hesj@cse.kyutech.ac.jp

Abstract

Many industrial robots are found in practical use in recent years. In principle, robots are simply repeating the operations taught by skilled human operators. Since the repetition is normally very accurate, the productivity in terms of the amount of products has been greatly improved. However, with the increasing demands on less amount but greater variety of the products, teaching a new sequence of operations to a robot is constantly required. Teaching to a robot requires not only teaching but also verification. The existing methods for teaching, direct or indirect, are cost-ineffective in terms of sensor requirements and verification costs. This paper presents a smart teaching method with which both teaching and verification can be done in a virtual space. The basic idea behind is to calibrate and model a real-world robot and generate its virtual counterpart. Teaching to the virtual robot is done with a data glove and a magnetic location sensor. Our experiments have verified that direct teaching to a robot in a virtual space has many advantages over the direct teaching to a real-world robot.

Key words: Virtual Reality, Direct Teaching, Robot Arm, Mechanical Assembly, Data Glove, and Magnetic Location Sensor

1. Introduction

Our past efforts have concentrated on solving task-planning problems through the understanding of assembly instructions and illustrations [2], [3], [4]. At that time, our target was to come up with a system for the automated assembly. In this paper, we report a method for direct teaching to a robot arm in a virtual space for the implementation of tasks such as assembly operations.

With the application of industrial robots, the productivity in terms of the amount of products has been greatly improved in recent years. This is because the industrial robots can repeat whatever they have been taught with extremely high accuracy. However, with the increasing demands on less amount but greater variety of the products, the operations for a robot to execute are constantly changing and hence teaching and verification of a new sequence of operations are frequently required.

The existing methods for teaching are divided into indirect ones and direct ones. The indirect methods assign to a robot a sequence of numerical data such as the coordinates and angles of the joints on the robot. Since a different operation corresponds to a different set of numerical data and sometimes a single operation may correspond to several sets of numerical data, the frequently required repetition of the teaching process is cost-intensive in terms of the numerical data acquisition. Moreover, subtle errors in the numerical data may cause the collision with obstacles and thus may fatally damage the robot in the process of verification. Vision-based indirect teaching approaches have been challenged [1], [5]. Regrettably, with the poor reliability of the vision processing results, it is very difficult for a robot to correctly recognize the operations performed by human operators.

Different from the indirect methods, direct methods focus on the interaction between human operators and the robots. One of the past efforts is to hold a real-world robot arm through a virtual data glove and let the real-world robot arm follow the motion of the virtual data glove [6]. Since the motion of the real-world robot arm is monitored through a video camera at a fixed location, only a limited set of viewpoints are available and thus teaching of the operations requiring high precision cannot be expected. Another direct teaching method is to directly move the robot arm in the real world, which requires the robot arm be equipped with a force torque sensor [7]. With this method, a human operator needs to hold the robot arm with his hand in the whole teaching process. It is, therefore, impossible to teach some special operations such as insertions in a narrow space and of course impossible to teach to a robot which is located in a dangerous place.

This paper presents a completely new method with which both teaching and verification are done in a virtual space. The basic idea is to calibrate a real-world robot arm and model the robot arm in a virtual space. Teaching is directly done to the virtual robot arm through a data glove and a magnetic location sensor. Since the operator's hand is also modeled in the virtual space, grasping the gripper of the virtual robot arm is recognized through the detection of the interference between the hand model and the gripper of the virtual robot arm. The advantages of our new method are as follows:

(1) Both teaching and verification are free of collision with obstacles;

(2) The tasks that has never been achieved with the force torque sensor based direct teaching method can easily be accomplished, because human operators need not to appear in the robot's workshop in the whole process of teaching and verification and the virtual hand model is visually erasable as needed;

(3) Human operators can change their viewpoints as they like and thus more correct results of teaching and verification can be expected;

(4) No force torque sensor is required throughout the whole teaching process;

(5) The movement of the virtual robot arm is smooth and fast;

(6) The refinement of the teaching contents, e.g., removing the redundant points recorded during the teaching process, can be done through the repetition of the no-cost verification process;

(7) A huge and heavy robot arm working with huge and heavy objects in the real world can also be taught because their virtual counterparts are weightless and scalable.

The rest of this paper is organized as follows. Section 2 describes how to model a robot arm and the operator's hand. Section 3 gives the details of our teaching and verification methods. Three experiments are done and the results are shown in Section 4. Section 5 concludes the paper with discussions about the future work.

2. Virtual World Modeling

In order to verify our idea of direct teaching to a virtual robot, we need to generate a virtual world. Currently, our virtual world consists of two virtual objects, a virtual robot arm and a virtual model of the human operator's hand. This section discusses how to model the robot arm and the operator's hand in the virtual space.

2.1 Robot arm modeling

Robot arms are widely used in various fields. The classification of the robot arms is normally based on the size, function, and application purpose. Figure 1 shows one of the pioneer robot arms, the MOVEMASTER, which is selected for verifying our idea of direct teaching to a virtual robot arm. The construction of the robot arm and the parameters related to our experiments are depicted in Fig. 1.

As shown in Fig. 1, the MOVEMASTER is composed of nine components: base, waist, body, upper arm, forearm, wrist, hand, and two fingers. Our steps to the modeling of the MOVEMASTER are (1) modeling of the individual components and (2) assembly of the modeled components. Figure 2 shows the model of the MOVEMASTER. As shown in Fig. 3, a hierarchical structure is used to organize the nine components. With the hierarchical structure, the behavior of the virtual robot arm is completely determined, i.e., the components in the bottom layers move together with the components in the layers above them.

2.2 Hand modeling

With the same modeling method, the operator's hand can be modeled as well. Figure 4 shows the hand model, which is composed of 19 components. The behavior of the hand model includes 3D movement in the virtual world and the hand gestures generated by bending or stretching finger joints alternatively. The former is detected with the magnetic 3D-location sensor. The latter are recognized with a data glove. Note that the magnetic location sensor and the data glove are the only means for the interaction with the virtual robot arm.

3. Direct Teaching in the Virtual World

The overall process of teaching consists of five steps. The first step is the capturing of the sensor information such as the 3D location and the gesture of the operator's hand model. The second step is the processing of the sensor information. In the case that the hand model is found close enough to the robot's hand component and is in a grasping gesture, the robot's hand component and the operator's hand model will be bound together. Otherwise, it will go back to the first step for the updated sensor information. The third step keeps capturing the information of the 3D location of the operator's hand model and passes the information to the fourth step. Since the robot's hand component has been bound together with the operator's hand model, the 3D location of the operator's hand is considered the goal location of the robot's hand component. Any updated information will be passed to the fourth step. At the fourth step, all the joint angles will be computed. The fifth step checks whether the joint angles obtained at the fourth step are valid or not. If they are valid, it will go on to the sixth step, at which the robot arm moves by changing the joint angles appropriately. Otherwise, it will go back to the first step.

This section addresses the following problems in detail.

(1) how to compute the joint angles,

(2) how to detect the grasping of the robot arm with the operator's hand model,

(3) how to detect the grasping of virtual objects with the virtual robot arm,

(4) how to constrain the motion of the virtual robot arm,

(5) how to use auxiliary lines to improve the accuracy of assembly operations.

3.1 Computation of joint angles

Similar to the real-world robot, a virtual robot arm changes its 3D position also by changing the angles at its joints. Figure 5 shows the geometrical relationships which are employed for the computation of the joint angles, θ_1, θ_2, θ_3, and θ_4. The origin of the coordinate system in Fig. 5 is located at the center point of the joint between the components, upper arm and waist. L1 is the length of the upper arm. L2 is the length of the forearm. L3 is the distance from the joint between wrist and forearm to the tip of the fingers. The coordinates (x, y, z) correspond to the tip of the center point between the two fingers. The computation of the angles, θ_1, θ_2, θ_3, and θ_4, is as follows.

$$x' = x - (L3*\cos\theta_5)\sin\theta_1 \qquad (1)$$

$$y' = y - (L3*\cos\theta_5)\cos\theta_1 \qquad (2)$$

$$z' = z + (L3*\sin\theta_5) \qquad (3)$$

$$A = \mathrm{sqrt}(x'^2 + y'^2 + z'^2) \qquad (4)$$

$$B = \mathrm{sqrt}(x'^2 + y'^2) \qquad (5)$$

$$\theta_1 = \mathrm{atan}(y, x) - \pi/2 \qquad (6)$$

$$\theta_2 = \mathrm{acos}((A^2 + L1^2 - L2^2)/(2AL1)) + \mathrm{acos}((A^2 + B^2 - z'^2)/(2AB)) \qquad (7)$$

$$\theta_3 = \pi - \mathrm{acos}((L2^2 + L1^2 - A^2)/(2L1L2)) \qquad (8)$$

$$\theta'_4 = \mathrm{acos}((A^2 + L2 - L1)/(2AL1)) \qquad (9)$$

$$\theta''_4 = \mathrm{acos}((A^2 + z'^2 - B)/(2A\, z')) \qquad (10)$$

$$\theta'''_4 = \mathrm{acos}((L3^2 + (L3\sin\theta_5)^2 - ((L3\cos\theta_5)^2\sin^2\theta_1 + (L3\cos\theta_5)^2\cos^2\theta_1))/(2L3^2\sin\theta_5) \qquad (11)$$

$$\theta_4 = \pi - (\theta'_4 + \theta''_4 + \theta'''_4) \qquad (12)$$

3.2 Detection of grasping with the operator's hand

Direct teaching requires the operator grasp and move the robot arm directly. It is, therefore, necessary to detect whether the operator has grasped the robot arm or not. The detection is done by checking the interference between the bounding box of the hand component of the robot arm and the bounding box of the palm component of the operator's hand model. If the interference is detected and the hand model is in the "grasping" gesture, it is concluded that the robot arm has been grasped.

3.3 Detection of grasping with the robot arm

A robot arm is expected to execute a variety of operations. As will be shown in Section 4, even the primitive operations such as pick-and-place and insertion require the robot arm grasp the virtual object. In order to detect the grasping using the same idea given in Section 3.2, we set a transparent box between the two fingers. The grasping detection is done with two steps. The first step is the detection of the interference between the transparent box and the bounding box of the virtual object. If the interference is detected, it is judged that the object is within the grasping range. At the second step, a pre-determined key is pressed to close the fingers and complete the grasping action. When the pre-determined key is pressed, the fingers will gradually move towards each other. The movement is realized by changing the 3D location of the fingers. As soon as the interference between either of the fingers and the bounding box of the virtual object is detected, the fingers will stop the movement.

3.4 Motion constraints

In order to let our virtual robot arm behave like a real world MOVEMASTER, we need to add a set of constraints to its motion. The constraints are based on the motion of the real world MOVEMASTER. It is well known that the motion of the real world MOVEMASTER is constrained to five degrees of freedom and each of them has a limited range of valid values.

Four of the five degrees are represented by the joint angles, θ_1, θ_2, θ_3, and θ_4, in Fig. 5. The fifth degree is represented by wrist rotation angle. As shown in Fig. 5, θ_1 is the waist rotation angle, θ_2 the angle between the upper arm and the X-Y plane, θ_3 the angle between the forearm and the upper arm, θ_4 the angle between the wrist and the forearm.

With the constraints, the virtual robot arm is always in the form of a straight line when viewed from above.

3.5 Auxiliary line based constraints

Assembly operations such as insertion of a shaft into a hole with very limited tolerance normally require the symmetrical axis of the shaft be aligned with the symmetrical axis of the hole with extremely high precision. Real-world robot arms are usually equipped with force torque sensors to overcome the alignment difficulties through trial and error based on the feedback information from the force torque sensors. In the virtual world, however, it is currently difficult for a virtual robot arm to get the force feedback. The alignment is, therefore, an unavoidable problem. In order to facilitate the alignment process, we introduce a concept, auxiliary line, which is conceptually similar to the line segments drawn for showing assembly relations in assembly illustrations. Different from the auxiliary lines in the assembly illustrations, auxiliary lines in this research are line segments in a 3D space. For an axis-symmetrical object, the auxiliary lines are attached to both its top and bottom and are collinear with its symmetrical axis. For an axis-symmetrical hole, an auxiliary line is set at the center of the hole and collinear with the symmetrical axis. The alignment is simplified by checking whether the auxiliary lines cross with each other. Whenever the auxiliary line of a shaft crosses with the auxiliary line of a hole, the robot arm holding the shaft will be compulsorily adjusted so that the two auxiliary lines become collinear with each other. Simultaneously, the motion of the robot arm will be constrained along the direction parallel to the auxiliary lines.

In principle, as long as the robot arm has sufficient degrees of freedom, the auxiliary line based constraints might be applied to the assembly parts in any position. The virtual MOVEMASTER, however, only has five degrees of freedom. The auxiliary line based constraints are limited to the assembly parts whose auxiliary lines are perpendicular to a horizontal plane.

4. Experiments

4.1 Experiment 1: Pick-and-Place

The specification of Experiment 1 is as shown in Fig. 7. The Z-axis is set to be the waist rotation axis. The virtual robot arm is initialized as follows, $\theta_1 = \theta_2 = \theta_3 = \theta_4 = 0$. As shown in Fig. 7, three blocks (object1, object2, and object3) are used for the Pick-and-Place experiment. The three blocks are all cubes with sides, 30 mm, 20 mm, 15 mm, respectively. The centers of the bottom square of the cubes are located at (300, 200, 0), (300, 300, 0), (250, 250, 0), respectively. The experiment is to teach the robot arm to pick up object2 and place it on the top of object1 and then pick up object3 and place it on the top of object2. In this experiment, all the three objects are considered weightless and thus all the unexpected happenings due to the weight in the real world can be ignored.

Figure 8 shows the teaching process. Figure 8 (a) shows the initial state. Figure 8 (b) shows the state, at which object2 has been picked and is being moved towards object1. Figure 8 (c) shows the state, at which object2 is being placed on the top of object1. Figure 8 (d) shows the state, at which object3 is being grasped. Figure 8 (e) shows the state, at which object3 is being moved and placed on the top of object2. Figure 8 (f) shows the goal state, at which object3 has been placed on the top of object2. Note that in the whole teaching process, the operator's hand model is erasable at any time as needed. The open and close of the gripper (the two fingers) of the robot arm is controlled with a key press.

4.2 Experiment 2: Pick-and-Place with alignment

This experiment is used to test the case in which force torque sensor is required for a real-world robot arm. The experiment is to pick a rectangular pole (object6) and place it at the corner composed by two plates (object4 and object5) with the same size.

The teaching process is as shown in Fig. 9. Figure 9 (a) shows the initial state. Figure 9 (b) shows the state, at which the gripper has been moved close to object6. Figure 9 (c) shows the state, at which object6 has been picked up and is being moved towards the goal position. Figure 9 (d) shows the state, at which the location of object6 is being adjusted towards the corner. Figure 9 (e) shows the state, which is considered the goal state based on the operator's recognition. At this state, the gripper is opened with a key press. Figure 9 (f) shows the bird view of the goal state. It can be seen from Figure 9 (f) that object6 is not very close to object4 and object5. The same experiment has been repeated ten times. The average distance between object6 and object4 is 3.2 mm. The average distance between object6 and object5 is 2.1 mm. Throughout this experiment, it might be concluded that for the pick-and-place with alignment, force torque sensor is not replaceable with the human vision.

4.3 Experiment 3: Insertion

This experiment is used to test a typical assembly operation, i.e., insertion. A cylinder with diameter, 2.8 cm and length, 3 cm will be inserted into a hole with depth 10 cm.

The teaching process is briefly shown in Fig. 10. Figure 10 (a) shows the state at which the cylinder has been picked up and the crossing between the auxiliary line of the cylinder and the auxiliary line of the hole is being checked. Figure 10 (b) shows the state, at which the two auxiliary lines have been automatically adjusted to be collinear. Figure 10 (c) shows the state, at which the cylinder has been inserted into the hole along the auxiliary line.

5. Discussions and Concluding Remarks

This paper presented a new approach to the direct teaching, with which both teaching and verification can be done in a virtual space. Our experiments have verified the advantages described in Section 1.

Throughout the experiments, on the other hand, the following disadvantages are found.

(1) The teaching is sensitive to the measurement errors of the 3D-location sensor and also sensitive to the movement of the operator's hand.

(2) The operations, which need force feedback information for a real-world robot can only be monitored with human vision and thus cannot be executed with sufficiently high precision in most cases.

(3) The weightless assumption may give rise to an incorrect teaching.

Our future work includes (1) to incorporate the weight effects and (2) to incorporate the force feedback into our current system by using the force torque sensor devices such as haptic master.

References

1. K. Kuniyoshi, et al., "Learning by showing", IEEE Trans. Robotics and Automation. 10, No.6, pp.799-822, 1995.
2. N. Abe and S. Tsuji, "Robot task specification in Natural Language", Proc. of Conf. of IEEE on Robotics and Automation pp.586-595, 1987.
3. N. Abe, K. Ohno, S. He, and T. Kitahashi, "Task Specification Using Technical Illustration", Proc. of Robotics and Automation, Vol.2, pp.58-64, 1993.
4. S. He, N. Abe, and T. Kitahashi, "On a System of Understanding of an Assembly Illustrations in Assembly Manuals", International Journal of Applied Intelligence, Vol.4, pp.367-382, 1994.
5. K. Ikeuchi, S. B. Kang, "Vision-based teaching to the robot hand", Journal of Robot Society of Japan, Vol.13, No.5, pp.599-602, 1995.
6. H. Ogata and T. Takahashi, "A Robot system that Learns Task and Executes in Different Environments Using Task Description Based on Geometry", Journal of Robot Society of Japan, Vol.11, No.3, pp. 444-452,1993.
7. T. Tomura, "A Study on Path Planning for Manipulator to avoid obstacles in Virtual Space", Undergraduate Thesis, Kyushu Institute of Technology, 1995.

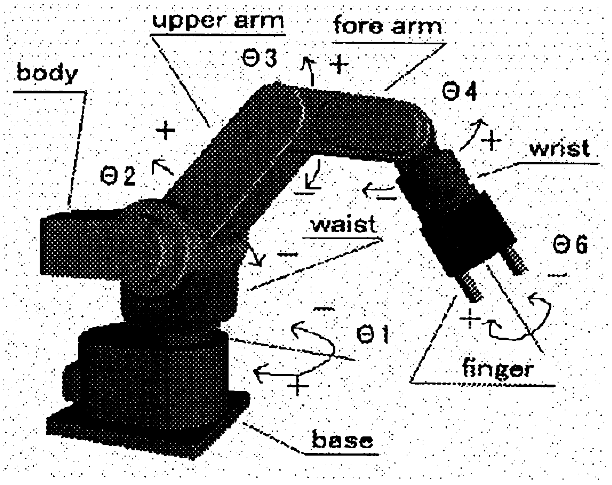


Figure 1: The MOVEMASTER and its construction.

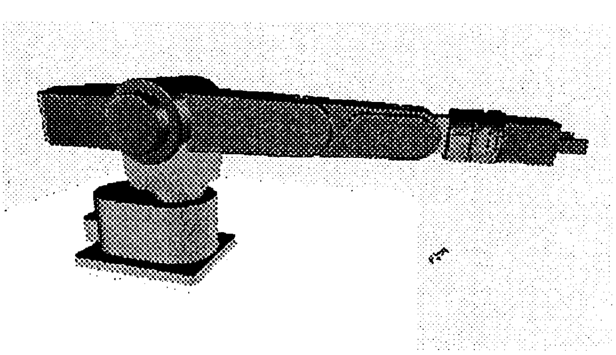


Figure 2: The model of the MOVEMASTER.

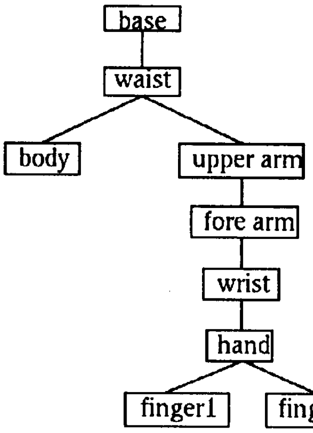


Figure 3: Hierarchical structure representation.

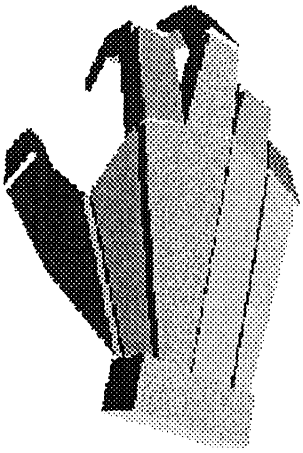


Figure 4: The hand model.

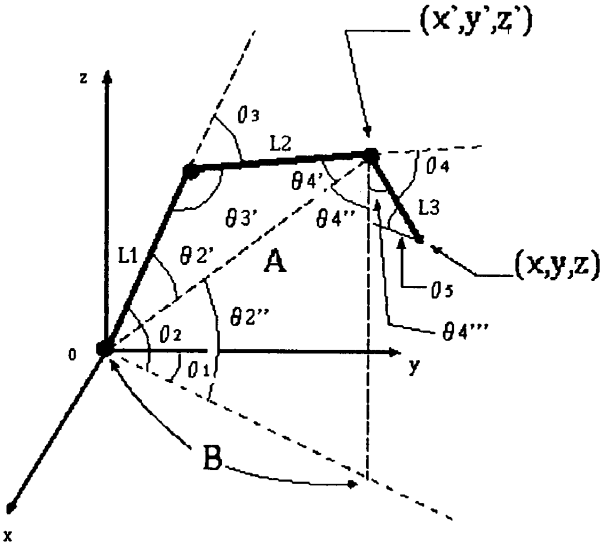
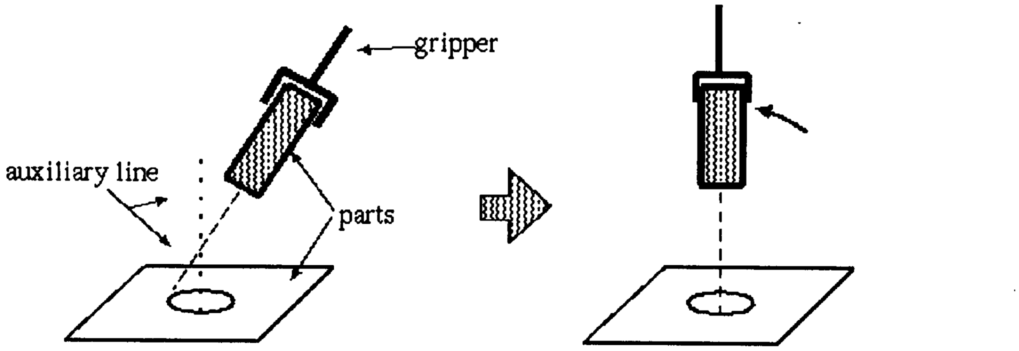
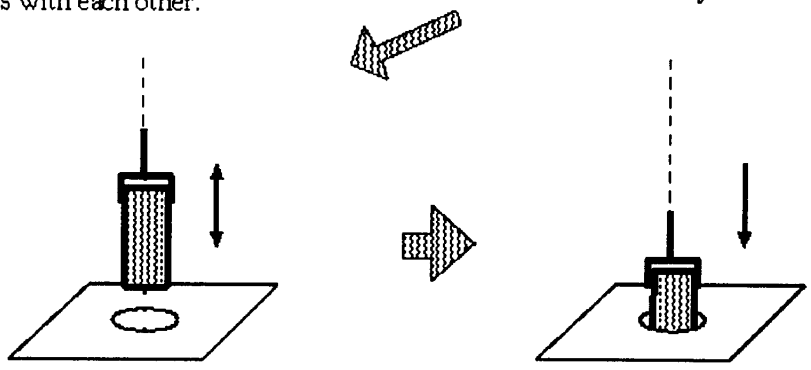


Figure 5: Computation of the joint angles.



Check whether the auxiliary lines cross with each other.

Compulsory adjustment of the robot arm so that the auxiliary lines are collinear.



Motion constraint: robot arm is only movable along the direction parallel to the aligned auxiliary lines.

Implementation of the insertion.

Figure 6: Auxiliary line based constraints for assembly operations.

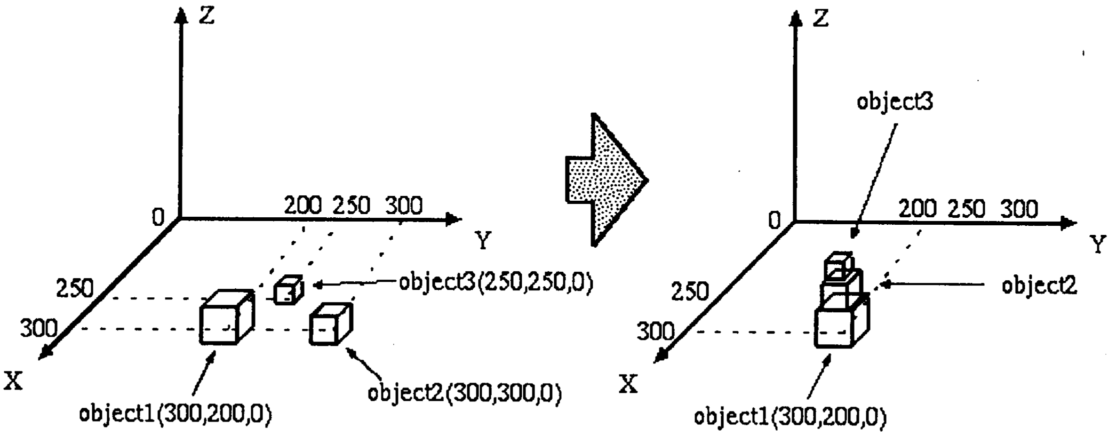
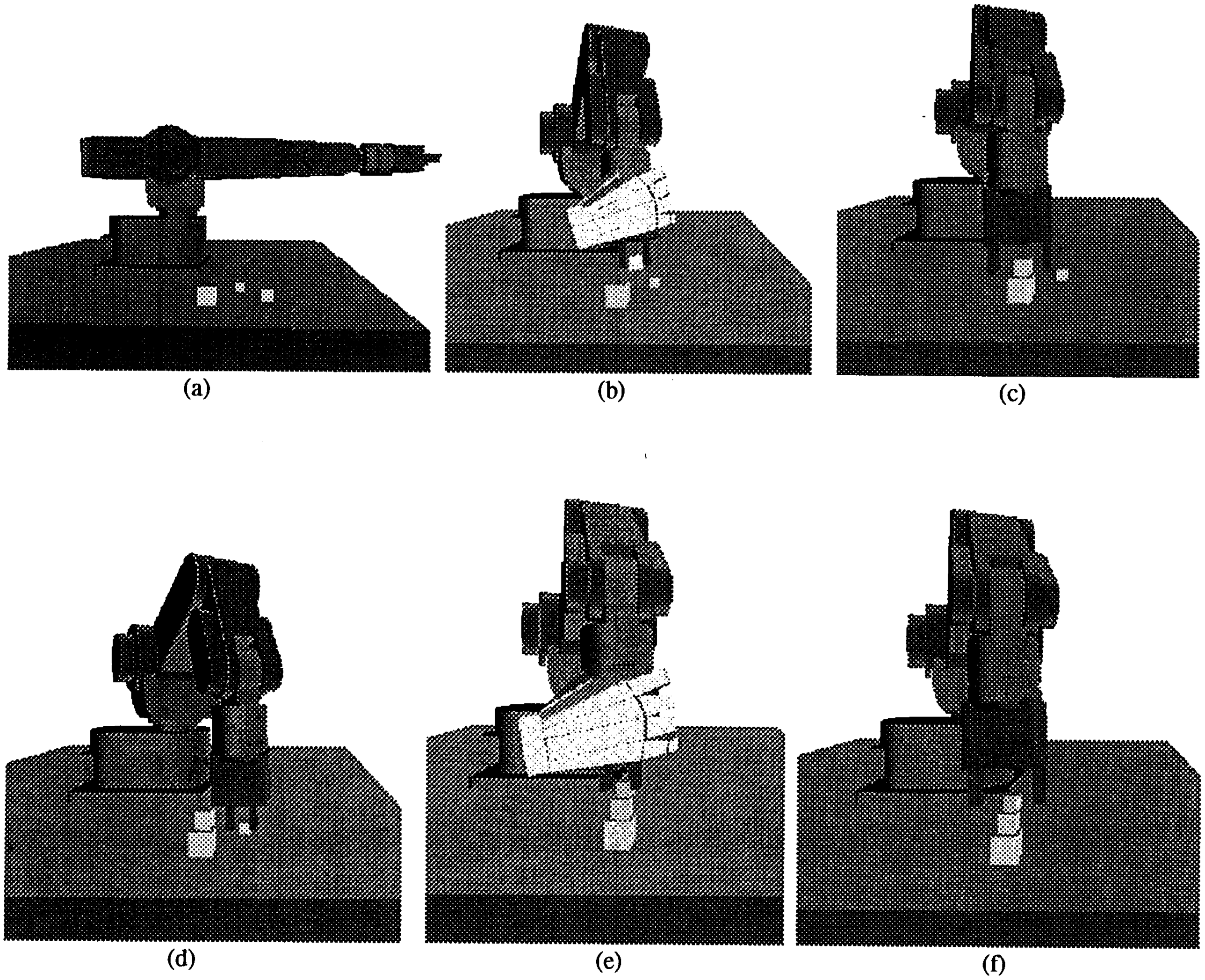


Figure 7: The specification of the initial and goal states of Experiment 1.


(a) (b) (c)

(d) (e) (f)

Figure 8: The teaching process for Experiment 1. (a) Initial state. (b) Intermediate state: object2 has been picked up. (c) Intermediate state: object2 has been placed on the top of object1 and the gripper has been opened. (d) Intermediate state: object3 has been picked up. (e) Intermediate state: object3 is being placed on the top of object2. (f) Goal state.

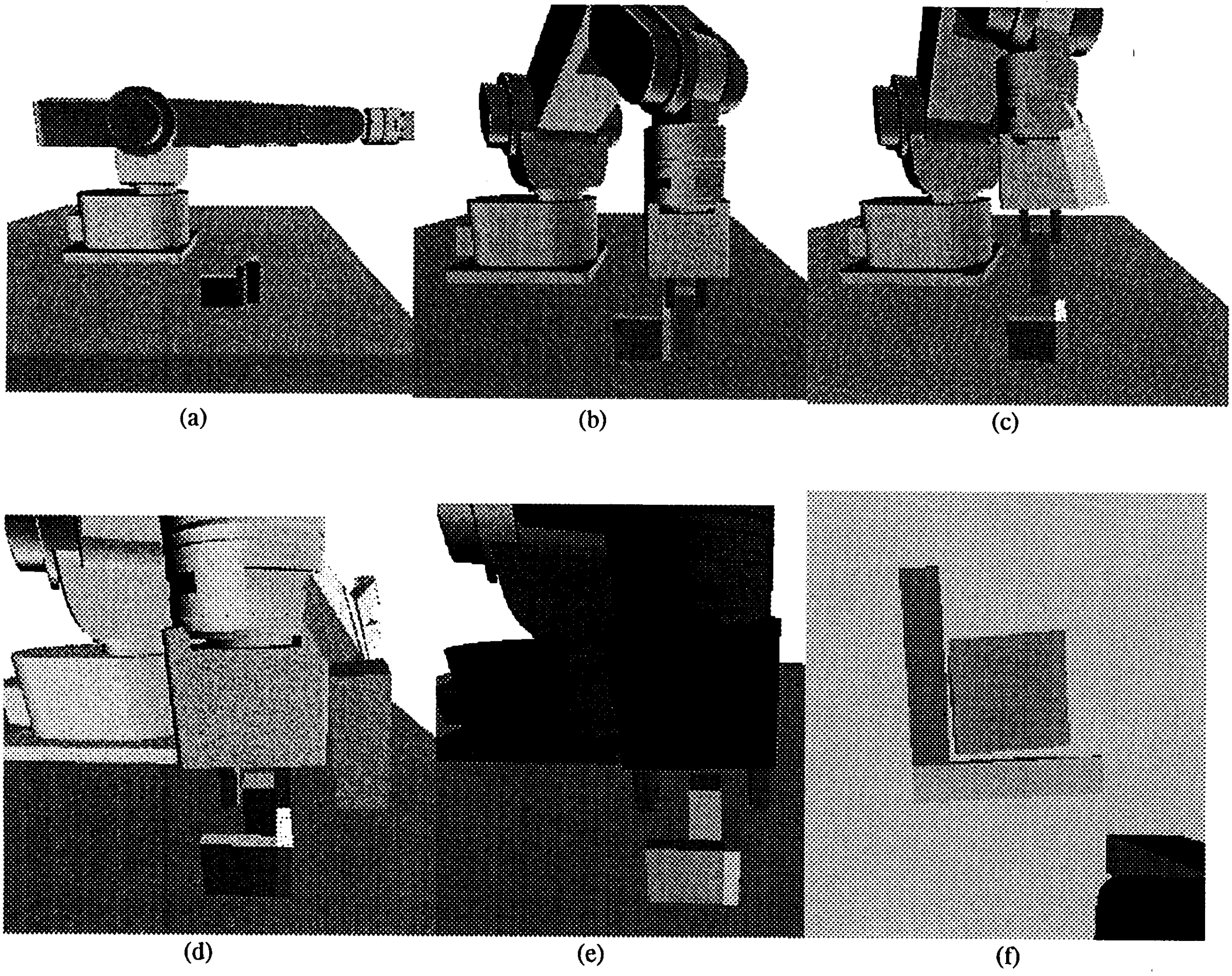
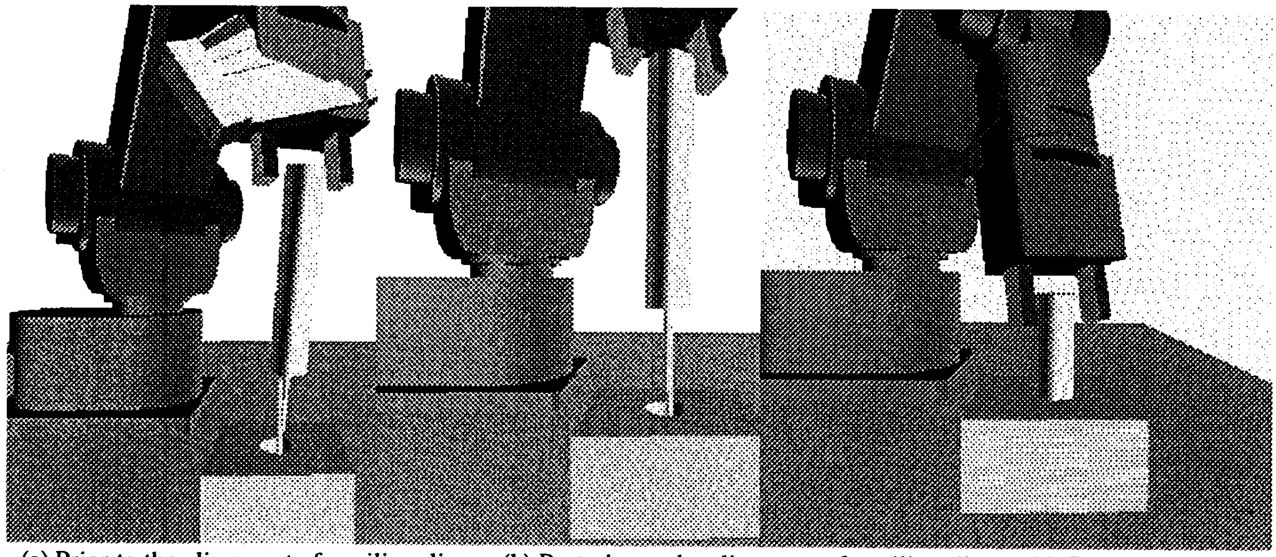


Figure 9: The teaching process for Experiment 2. (a) Initial state; (b) Intermediate state: object6 has been picked up; (c) Intermediate state: object6 is being moved towards the goal location; (d) Intermediate state: object6 is being adjusted to the corner composed by object4 and object5; (e) Intermediate state: object6 is supposed to be at the corner; (f) Goal state.



(a) Prior to the alignment of auxiliary lines. (b) Posterior to the alignment of auxiliary lines. (c) Goal state.

Figure 10: The teaching processing for Experiment 3.