N90-29888

MULTI-LEVEL MANUAL AND AUTONOMOUS CONTROL SUPERPOSITION FOR INTELLIGENT TELEROBOT

S. Hirai and T. Sato

Autonomous System Section, Intelligent Systems Division
Electrotechnical Laboratory, MITI
1-1-4 Umezono, Tsukuba-shi, Ibaraki 305 Japan

Abstract

Space telerobots are recognized to require cooperation with human operators in various ways. Considering the issue, this paper describes multi-level manual and autonomous control superposition in telerobot task execution. To realize the concept, we propose the object model, the structured master-slave manipulation system and the motion understanding system. The object model offers interfaces for task level and object level human intervention. The structured master-slave manipulation system offers interfaces for motion level human intervention. The motion understanding system maintains the consistency of the knowledge through all the levels which supports the robot autonomy while accepting the human intervention. The superposing execution of the teleoperational task at multi-levels realizes intuitive and robust task execution for wide variety of objects and in changeful environment. The performance of several examples of operating chemical apparatuses is shown.

1. Introduction

A telerobot is a highly integrated system. It should be able to execute task autonomously and at the same time should be able to cooperate with human operators in multi-level. The telerobot with such requirements should cover wide spectrum of technology from AI to manipulator control. Not only the development of various component for the telerobot but also the integration scheme of them should be well considered.

The first stage of teleoperator development was providing remote manipulators with autonomous functions. The accumulation of task repertories and related task data was the basis of such autonomous functions [1-4]. Recent telerobots [5-8] and telerobotic systems [9-11] incorporate knowledge base or world model. While the effects of accumulated autonomous functions have been recognized in a general sense, the framework for the knowledge base and the world model has been paid less attention from robotics researchers. The point is that the framework should take into account not only simple data and procedures to handle objects but also infrastructural functions for cooperative execution of tasks. The **cooperative execution framework** considering the nature of telerobot tasks is required.

Cooperative execution of telerobot tasks is necessary for several reasons [7-8,12-13]. Robots are not enough intelligent that they cannot make complete plan of tasks. So, human operators should help by expanding tasks into sequence of elementary operations. This is the task level intervention. The object level intervention, lower than the task level one, is also necessary because robots cannot recognize environment and objects as human operators can do. So, human operators should help robots by teaching.

Robots needs further intervention at the lowest level, i.e. at the manipulator level. It is the case robot cannot generate skillful motion to do the task and is very different from intelligent intervention usually incorporated in the conventional supervisory controls [12]. We need further consideration on this issue.

Autonomous functions provides repetitive execution of task. On the other hand, fixed set of task repertories lacks adaptability to even small changes of task conditions. It is also difficult to apply fixed task repertories to a new environment because most data needed for task programs are not available. Direct maneuvering of a telerobot is suitable to execute task immediately, while repetitive task execution is tedious and it is difficult to control a fine and compliant motion from the remote site because of the degraded communication channel. Since space telerobots feature a wide spectrum in both task environment and communication capability, there should be a new cooperative way of teleoperational task execution between a robot and a human operator, i.e. **superposition of autonomous function and direct maneuvering.**

In addition to this, utilization of both the autonomous functions and the direct maneuvering generates another problem specific to telerobots. It is the matter of consistency maintenance of the robot knowledge while accepting the multi-level human intervention. On of the most important and recent topics related to this issue is the world model management [6-8]. As far as tasks are executed through the autonomous functions of telerobots, automatic keeping of the world model with the real environment data is possible by introducing world model maintenance instructions as discussed in this paper. However, if the tasks are executed through the direct maneuvering of a human operator, those maintenance instructions will not work. The **automatic management of the world model** while accepting the human intervention is required.

The Model Enhanced Intelligent and Skillful TElcRobot (MEISTER) [15], being developed at the Electrotechnical Laboratory, is an integrated test bed to study the multi-level cooperation between man and a robot. The MEISTER features the object model which works as the framework to accumulate task repertories with integrity, the structured master-slave manipulation system which can superpose both autonomous task execution of robots and direct maneuvering of a human operator, and the motion understanding system to realize the world model management even for the direct maneuvering. In the followings, we present prototype of the MEISTER and discuss those features suitable to space telerobots.

2. Object Model Suited for Telerobot [15]

Compared to the industrial robots on production lines, the telerobots are required more flexibility with respect to the task conditions. Objects are moved and their relationships in environment changes dynamically. Objects are to be handled even if some data are indeterminate yet in emergency. Procedures to handle an object should be arranged according to the changeful environment.

2.1. Outline of Teleoperation Using Object Models

Figure 1 outlines a telerobot system equipped with object models. Each object model contains knowledge to execute tasks. To start a task to handle an object, the operator gives a command of the task to the corresponding object model. Then the model supports the task execution.

The model of an alcohol lamp, for example, contains knowledge such as procedure for lighting and data of the wick point. Procedures and data for pick and place tasks are also available as general knowledge through hierarchical modeling system. The operator may command the model of lamp to light it. When all the information necessary for the task or the environment satisfies conditions to start the task, the lighting procedure is executed by the robot automatically. If the model lacks any data necessary for the task, it invokes special procedure in which the model acquires the data while the operator executes the task. The operator can also command a model to execute a part of the task or from the middle of it. The purpose of the teleoperation with object models is to enable the utilization of autonomous functions with immediacy and flexibility in this way.

2.2. Policies for Construction of Object Models

In such teleoperation style the object models offer the following features.

a) Knowledge for a task to handle the object are described, stored and accessed easily.

b) Task flow is changed according to the environment state.

c) Maintenance of consistency between model data and real environment is done automatically.

d) Knowledge for primal tasks in teleoperation such as pick and place is prepared.

e) Ta k can be started quickly even when model data is incomplete.

In the construction of the MEISTER system, we have introduced three policies for the object model to be suited for teleoperation.

Making a module of handling knowledge object by object: In contrast with collecting handling knowledge based on procedure types, the knowledge is collected based on the types of objects to be handled. It brings about the following merits:

a) Since a variety of task knowledge using an object are packed in the same framework, they can be retrieved intuitively by specifying the name of the object.

b) Changing task flow according to environment becomes easy. Since information representing the state of the object in the environment is stored object by object, not distributed over the whole programs, the state of the object can be checked easily.

c) Management for model data changing along with task execution becomes easy. Information and procedure to handle an object is stored in the same framework. Management process can also be stored in the same framework. These two features of the object model enables to generalize and concentrate the management process.

Define "general handling model": Pick and place tasks are fundamental and frequent in teleoperation. They are essential operations in object handling tasks. Either of them is composed of the same motion patterns for wide variety of objects. Therefore we define a general model of handling objects called 'general handling model' to describe the properties and procedures common to all the objects as a target of pick and place tasks. On the other hand, a model for the class of specific object like an alcohol lamp is called a 'specific object model.' Figure 2 illustrates the hierarchical relationships among the general handling model and specific models. Typical properties and procedures of the models are listed in the figure.

By definition, properties and procedures of general handling model are shared by all the object models. It corresponds to the inheritance mechanism of the object oriented system by which we have implemented the models. Introduction of this model hierarchy brings the following merits:

a) Defining a specific object model, such as an alcohol lamp, to be a specialization of the general handling model, it becomes automatically the target of pick and place tasks. Consequently, fundamental object handling procedures stored in the general handling model becomes available in specific object models without any additional definition. Sharing general knowledge among the general object and specified objects saves the space for memorization.

b) It gives unity in management of environment state changing with task execution. Environment state here means object pose and affixment relationship among objects. Basically, pick and place tasks change the environment state. In other words, moving an object is mainly commanded by fundamental motion commands which constructs pick and place operation. Basic model data management is concentrated if the management procedures are built in the general handling model with pick and place related procedures.

Introduce teaching-executing method: As pointed out before, objects are to be handled even if some data are indeterminate yet in emergency. To cope with the situations, we introduced a teaching-executing method [4]. In this method, the model checks the availability of data. If it is not available, the model requests the operator to control the robot for the execution and the model acquires the data from the executed task. The method brings about the following merits:

a) It enables to start a task quickly by direct manual control, and also to execute the task automatically from the next time using the model data taught through the manually executed task.

b) There are cases where task executing motions themselves are not appropriate for model data teaching. Special teaching procedures for such cases can be possessed by the model. Using these procedures, the teaching-executing method realize reliable acquisition of the environmental data.

The object models of the MEISTER allow object level intervention by the teaching-executing method. They also allow task planning level intervention by macro expansion capability of tasks and confirming execution of the expanded procedures.

3. Structured Master-slave Manipulation System

To allow manipulator level intervention in the MEISTER system, we provide the structured master-slave manipulation system. As is going to be made clear in the followings, 'structured' means that our system offers formalized motion patterns which can be superposed on the conventional master-slave control without changing control mode.

Figure 3 shows the functional block diagram of the structured master-slave manipulation system. The available commanding devices are the master manipulator, a teach pendant, a keyboard and the time dial. Control schemes to be superposed on the conventional master-slave control are as follows.

Resolved motion rate control scheme [16]: The slave manipulator is moved in the selected axis of the named coordinate system at a specified velocity while the button of the teach pendant is pressed. The scheme is suited to realize precise linear motion either in joint angle space or in Cartesian coordinate space.

Incremental control scheme: The x, y and/or z position of the slave manipulator can be incremented in a named coordinate system at a specified amount when the operator strikes the corresponding key. Increments of rotation in each axis can be commanded in the same way. It has been reported that the scheme offers higher precision than the master-slave control scheme if signal transmission delay exists between the master and the slave [17].

Indexing scheme: The coordinate transformation between the master and the slave manipulator is calculated every time when the master-slave control scheme is initiated. It enables to set the slave manipulator at a pose suitable for the task execution and the master at a pose comfortable to the operator [18]. This function also virtually expands the movable range of the master manipulator.

Software jigs [4]: A software jig describes specific motion constraint superposed on the slave manipulator motion. It has the effect of a hardware jig, but is specified by a special software package. Consider the task of carrying a glass filled with water. The operator must focus his attention on maintaining the orientation of the glass vertical in order not spill the water. This constrained motion increases the working load of the operator and the difficulty of the task. The software jig supports the motion constraint while the jig is active and enables the operator to concentrate on only its positional movements.

Skill superposition scheme [19]: In software jigs, the robot supports fixed constraint motions in a task motion. However, some tasks requires more complicated control like compliant motions. Typical one is a peg in the hole, where the orientation of peg should be compliant to the hole. Even a simple placement of an object on a table requires to guard not to hit the table hardly and not to incline the object too much. The skill superposition scheme of the MEISTER supports control for these secondary motions while leaving the operator the control of whole task progression.

Figure 4 explains the effect of this scheme in a task to pull a peg out of a hole. Employed skill here is that the robot keeps the peg compliant in orientational motion. The operator controls movement parallel to the axis. The effect of the scheme is clear at the early trials of the experiment. The difference of achievement with and without the scheme decreases along with the trials. It is because the operator has acquired the skill. This shows that the skill superposing scheme helps a novice operator to achieve skillful operations.

Programmed control scheme [14]: When the task is controlled by a program, the slave manipulator moves along the given trajectory. If the operator moves the master manipulator, the motion of the slave manipulator is modified correspondingly.

There are several possible realization of this scheme. The following is one example. If the master manipulator is moved such that the slave leaves a virtual tube specified by absolute value of radius as threshold concentric to the calculated trajectory, the slave manipulator goes to the master position/orientation instead of following the trajectory. When the master manipulator is not moved beyond the threshold, the slave tries to return to the nearest point on the trajectory. After the slave returns to the trajectory, it continues the programmed motion.

The time dial can be used to change the speed and direction of the program progression. Its potential applications are, teaching, monitoring and error recovery of robot tasks. For example, in monitoring of the task execution, slowing down the speed of the execution makes guarded motion easy. Reversing the direction of time is useful for error recovery. By specifying appropriate time point on the simulated execution on graphics, the operator can restart the task from the appropriate step or he can skip needless sub-tasks easily.

4. Motion Understanding for telerobot

4.1. Management of World Model

So far as the robot is commanded a task with higher level instructions which include necessary instructions to update the change of the world model, it can maintain the consistency of it. On the other hand, it becomes impossible for the robot to maintain the consistency if the real world is changed by commands or events other than such instructions. Typical example is when the operator intervenes at the motion level. However motion level intervention is inevitable in telerobot tasks as discussed previously. Therefore we have developed the motion understanding system which recognizes the meaning of the operated motion [15]. Recognized results are used for the world model maintenance. Thus the motion understanding system maintains the consistency of the robot knowledge of all the level while accepting the human intervention.

What kind of task motion should the system understand? There exists wide variety of tasks to do for telerobot such as machine assembly, tool operations and so on. Among these task motions, the followings concern about the model management stated above; grasp motion, move motion, attach motion and set motion. These are so called pick and place task related motions. Since pick and place is the most basic robot task which is closely coupled with the model management, we focus on the motions for a pick and place task.

4.2. System Architecture

The block diagram of the motion understanding system is shown in Fig. 5. Symbolizers continuously monitor the signals such as pose and finger force of the remote manipulator, extract specific task events such as closing of the finger, and convert them into predefined symbols. Detected events are sent to the motion understanding interpreter. Cell manager realizes efficient processing by dynamically limiting the target objects to only those close to the manipulator hand. The interpreter tries to recognize the meaning of the remote manipulator motion by matching the events from symbolizers with the pre-state and post-state of the task models expressed in the form of rules. The Rule Base contains the rules.

While symbolizers work at the ratio of servo cycle of master-slave control, the interpreter works when the symbolizers report the events. The separation of processing level in this manner makes the system hierarchical consisting of the symbolizer and interpreter layers. It contributes the efficiency of overall processing because the interpreter, which is more complicated and time consuming procedure than that of symbolizers, runs only when events are reported. It also enables the notation of rules to concentrate on only the processing level of the task model description and contributes the readability of the motion understanding rules.

4.3. Experiment

The conducted experimental task is to transfer an alcohol lamp on a table from one place to another using the master-slave manipulator. Figure 5 shows the task environment and the loci of the robot hand: The alcohol lamp at position A is transferred to the point C over the match box B.

The system continuously monitors the distance between the coordinate center of the robot hand and the target object. At position 1, the system recognizes the approach motion by detecting that the hand comes close to the lamp. At position 2, the system recognizes that the robot hand is out of the approach region of the lamp and is in the grasping region. At the same position, the grasp motion is recognized when the system detects the closing of the slave fingers and the increment of grasping force. After recognizing the grasping of the lamp, the system generates an affixment relationship between the slave hand and the object. Hereafter the system can keep the current value of the lamp pose by monitoring the movement of the slave hand. At position 3, the system detects that the lamp is pushed to the table and then detects that the robot hand released the lamp.

This experiment shows that the system can recognize a pick and place motion of the human intervention using a master-slave manipulator. It also shows that the consistency between world model and real world can be maintained by the results of motion understanding system.

5. Demonstration of MEISTER system

As the benchmark of the whole MEISTER system, handling chemical apparatuses is selected. Figure 7 shows the set up of the working environment. Using these apparatuses, we have been testing the object models, the structured master slave manipulator and the motion understanding system. Typical operations and functions of the MEISTER employed are as follows.

In picking up the spoon, stored pick and place procedures and teaching-executing method of the general handling model are employed.

In scooping up sample material with the spoon, special motion for scooping up defined in the model of a spoon is employed.

In carrying the sample on a paper on the balance and carrying the paper containing the sample to the bowl, a software jig to keep the posture of the spoon and the paper is employed.

In grinding the sample, programmed motion to move the pestle on the circle trajectory is used. In the midst of the grinding, program superposition scheme is employed to crack a specific particle by the intervention through the master manipulator.

In rearranging the alcohol lamp position, the human operator uses the master manipulator. This operation is recognized by the motion understanding system and the model data of the lamp is updated automatically.

In lighting the alcohol lamp, special task procedure defined in the lamp model and the updated data is used. In striking the match, special procedure defined in the match model is called.

Returning operation of the spoon are basically executed by the robot automatically using the data taught in the teaching-executing method in pick up operation of them.

The rest procedures for a flame reaction experiment is executed in the same manner. These operations and employed schemes of the MEISTER show the variety of telerobot tasks and the usefulness of multi-level cooperation of man and the robot.

6. Conclusion

This paper presented the Model Enhanced Intelligent and Skillful TEleRobot (MEISTER). It consists of knowledge base on the object models, the structured master-slave manipulation system and the motion understanding system. These three components form a trinity to realize multi-level human intervention and task cooperation in telerobot tasks.

It is quite natural for the space telerobot, which will achieve variety of missions in space station, shuttle and so on, to have handling knowledge about objects and execute tasks autonomously based upon it. Flexible modification capability of robot task execution at multi-level will contribute wide application of telerobots to material, biological, chemical experiments etc., and also in changeful task environment conditions in space.

The MEISTER is an integrated test bed to study the intelligence for telerobots.

7. Acknowledgment

The authors would like to express their great appreciation to Dr. T. Yuba, manager of the intelligent systems division of Electrotechnical Laboratory, and Dr. M. Kakikura, chief of the autonomous system section for the promotion of this research. They are indebted to Dr. K. Takase and Mr. T. Suehiro of Electrotechnical Laboratory for providing the ETA-II Direct-Drive manipulator as well as their invaluable advice on this research. They are also indebted to Dr. T. Ogasawara and Mr. T. Matsui for providing ETALisp and LEO, lisp and object oriented system developed for robotics research, with which the MEISTER system has been implemented.

8. References

[1] D. E. Whitney, "State Space Models of Remote Manipulation Tasks," Proc. 1st IJCAI, pp.495-507, 1969.

[2] W. R. Ferrell and T. B. Sheridan, "Supervisory control of remote manipulation," IEEE SpectrumOctober, pp.81-88, 1967.

[3] R. Fournier, P. Gravez and C. Mangeot, "High-Level Hierarchical Control in Computer Aided Teleoperation (CAT)," Proc. 87 ICAR, pp.411-420, 1987.

[4] T. Sato and S. Hirai, "Language-Aided Robotic Teleoperation System (LARTS) for Advanced Teleoperation," IEEE J. Robotics and Automation, Vol. RA-3, No. 5, pp.476-480, 1987.

[5] T. Sato and S. Hirai, "Design of Teleoperator System utilizing the World Model (in Japanese)," J. of the Robotics Society of Japan, Vol.4, No.4, pp353-362, 1986.

[6] J. S. Albus, R. Lumia and H.McCain, "Hierarchical Control of Intelligent Machines Applied to Space Station Telerobots," Proc. Space Telerobotics Workshop, Pasadena, CA, 1987, Vol. 1, pp.155-165.

[7] N. O. Sliwa and R. W. Will, "A Flexible Telerobotic System for Space Operations," ibid., Vol.2, pp.285-292.

[8] L. E. Parker and F. G. Pin, "Architecture for Task Allocation in a Man-Robot Symbiotic System," presented in 1987 SPIE Cambridge Symposium: "Advances in Intelligent Robot Systems,"

[9] T. Matsui and M. Tsukamoto, "An Integrated Teleoperation Method for Robots using Multi-Media-Display (in Japanese)," J. of the Robotics Society of Japan, Vol.6, No.4, pp.33-42, 1988.

[10] T. Hasegawa, T. Suehiro and T. Ogasawara, "Model-Based Integration of Environment Description and Task Execution, " NATO Advanced Research Workshop "CAD Based Programming for Sensor Based Robots" IL CIOCCO, Italy, 1988.

[11] Ph.Even and L.Marce, "PYRAMIDE:AN INTERACTIVE TOOL FOR MODELLING OF TELEOPERATION ENVIRONMENTS," Proc. IROS '88, Tokyo, 1988, pp. 725-730.

[12] L. Conway, R. Volz and M. Walker,TELE-AUTONOMOUS SYSTEMS:METHODS AND ARCHITECTURES FOR INTERMINGLING AUTONOMOUS AND TELEROBOTIC TECHNOLOGY, Proc. 1987 IEEE Int. Conf. on Robotics and Automation, pp. 1121-1130.

[13] V. Hayward, "AUTONOMOUS CONTROL ISSUES IN A TELEROBOT," presented at IEEE Systems Man and Cybernetics Conf., Pekin, China, Aug. 1988.

[14] T. Sato and S. Hirai, "MEISTER:A Model Enhanced and Skillful Teleoperational Robot System," in R.Bolles and B.Roth eds. Proc. 4th ISRR, pp. 155-162, the MIT Press, 1987.

[15] S. Hirai and T. Sato, "OBJECT MODEL FOR TELEROBOT," Proc. IROS '88, Tokyo, 1988, pp. 725-730. pp. 701-706.

[16] D. E. Whitney, "Resolved-motion rate control of manipulators and human prostheses," IEEE Trans., Man-Machine Syst., Vol. MMS-10, No. 2, pp. 47-53, 1989.

[17] G. P. Starr, "A Comparison of Control Modes for Time-Delayed Remote Manipulation," IEEE Trans. SMC, Vol. SMC-9, No. 4, pp. 241-246, 1976.

[18] S. M. Killough, H. L. Martin, H.L. and W. R. Hamel, " Conversion of a Servomanipulator from Analog to Digital Control," Proc. of IEEE Conf. on Robotics and Automation, pp. 734-739, 1986.

[19] S. Hirai, T. Sato and H. Urabe, "Teleoperation using Skill Superposition (in Japanese)," preprints for the Fifth Annual Conference of Robotics Society of Japan, pp. 99-100, 1987.

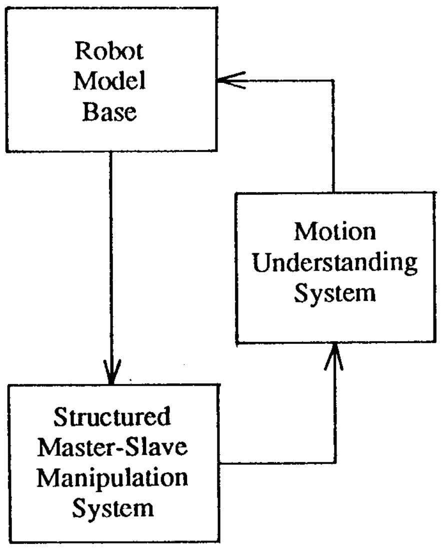

Fig. 1 MEISTER system architecture.

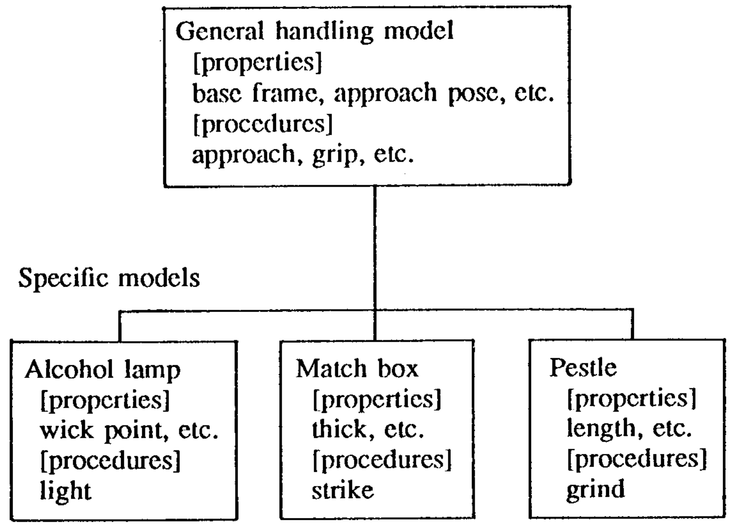

Fig. 2 Hierarchy of object models.

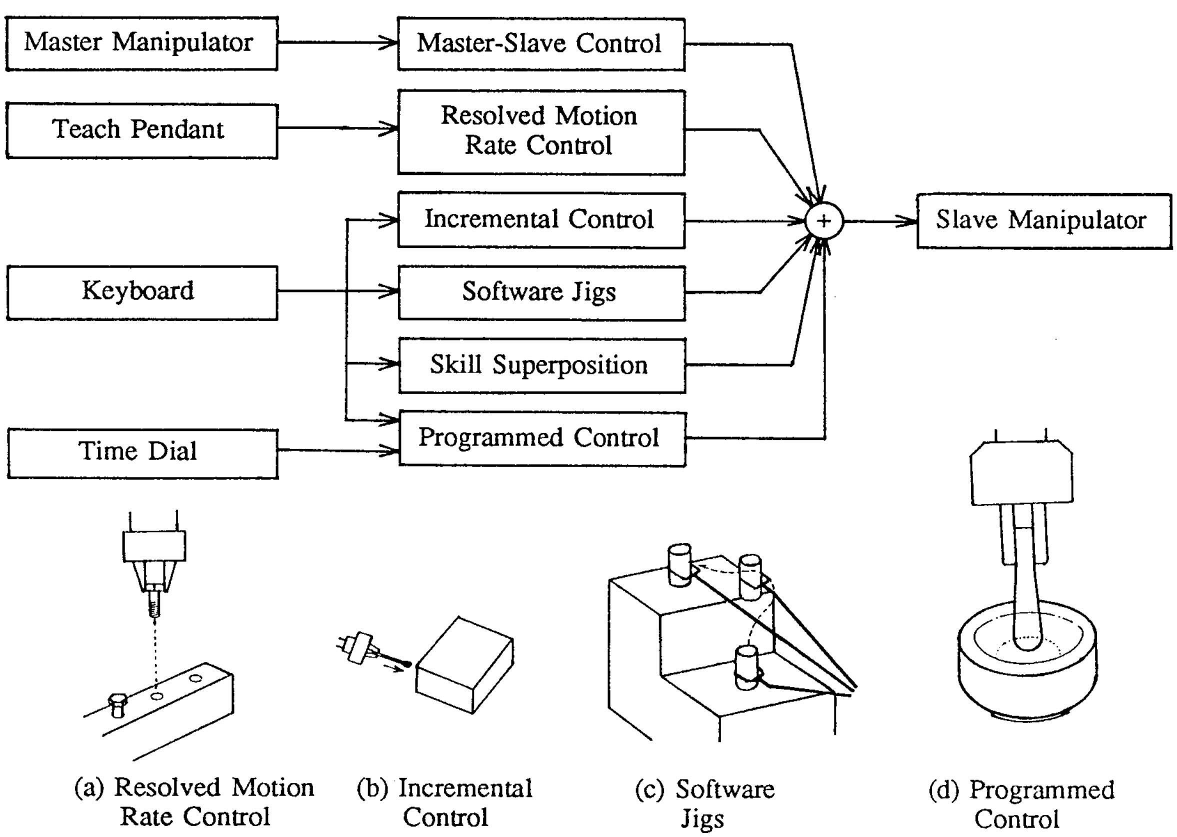

Fig. 3 Block diagram of Structured Master-Slave Manipulation System.

Table 1
Results from first to sixth trials.

Trials	1	2	3	4	5	6
without Skill	s	f	f	f	s	s
with Skill	s	s	s	s	s	s

s: success f: fail

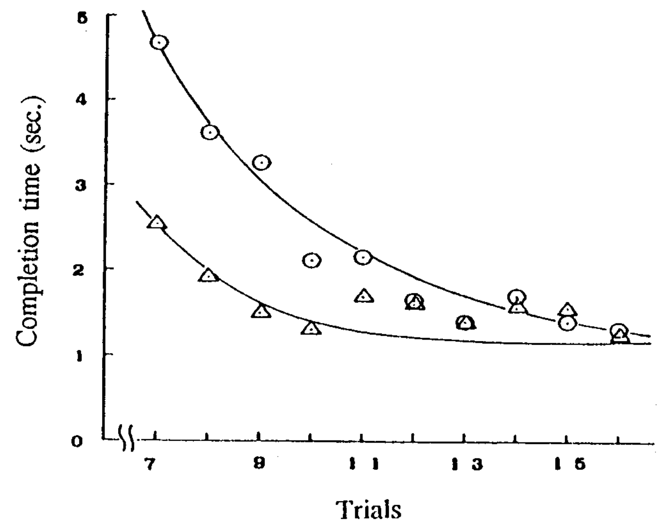

Fig. 4 Effects of skill superposition. "⊙" denotes result without skill, "△" denotes result with skill.

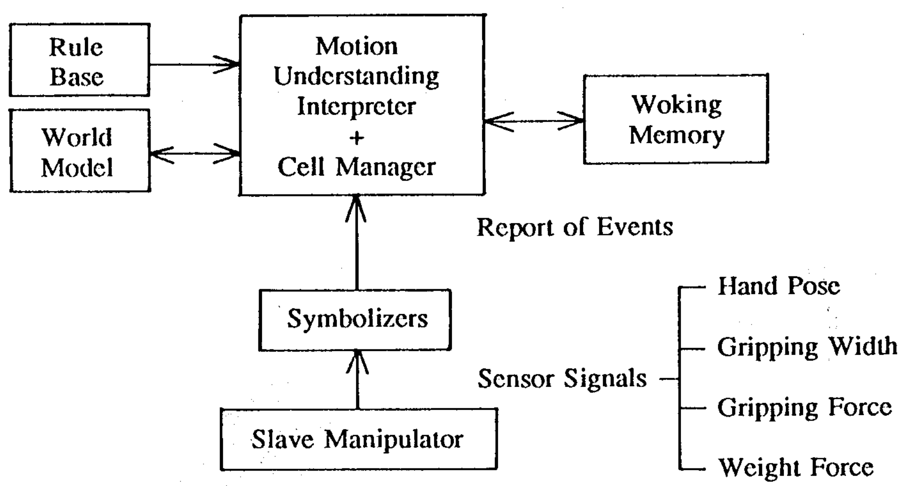

Fig. 5 Block diagram of motion understanding system.

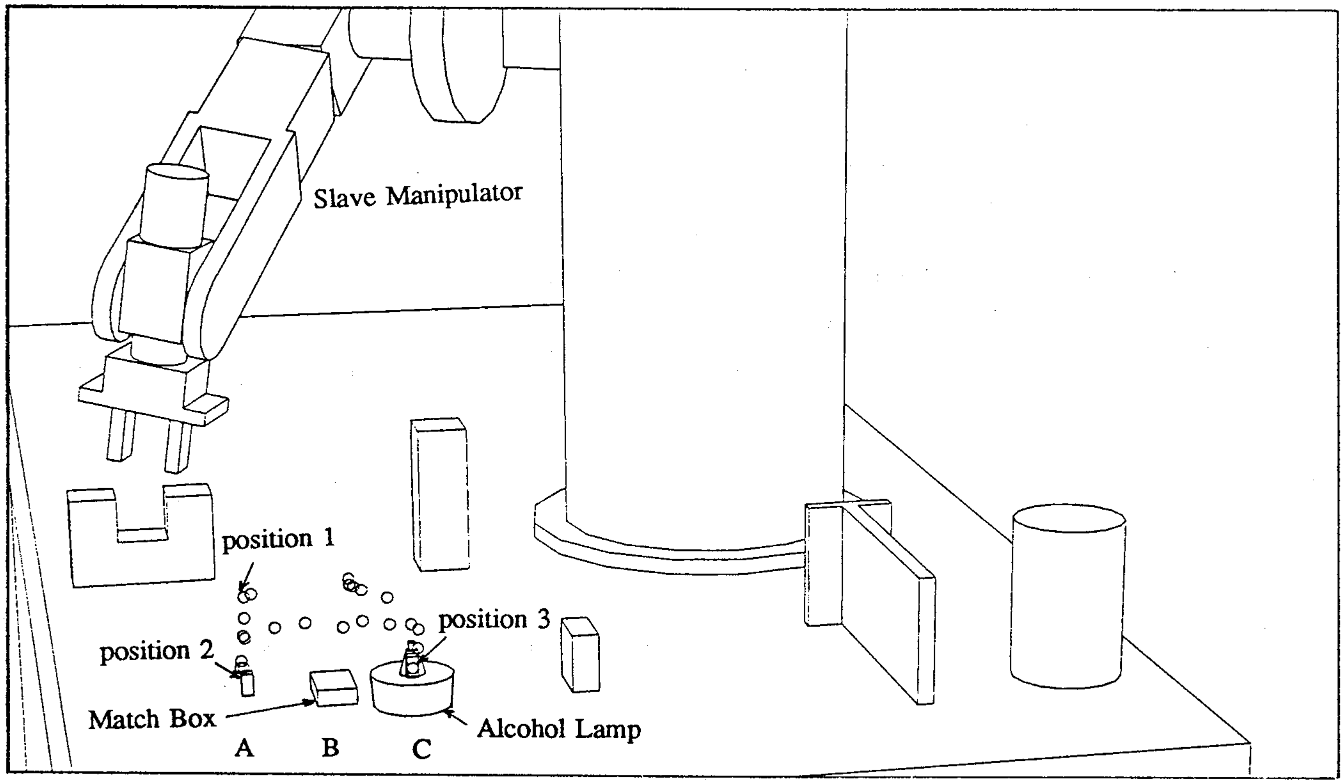

Fig. 6 Motion understanding experiment of carrying alcohol lamp.

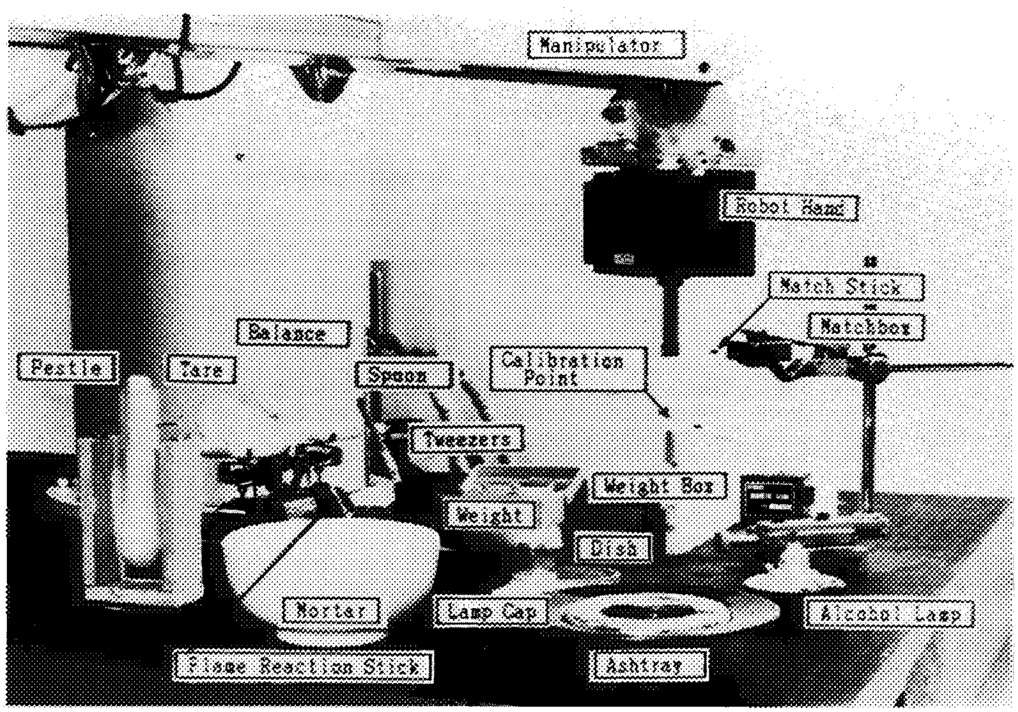

Fig. 7 Experimental working environment.