Noname manuscript No.
(will be inserted by the editor)

Extending Commands Embedded in Actions for Human-Robot Cooperative Tasks

Kazuki Kobayashi · Seiji Yamada

Received: date / Accepted: date

Abstract In this paper, we propose a novel method to control a robot without robot manipulation. Users do not need to precisely manipulate the robot and to learn the manipulation method. The proposed method can send commands to a robot by using a human action sequence that achieves their own task. In order to enable the robot to achieve tasks, we introduce a keep-based interaction in which a human keeps an action in the sequence for a certain period. The advantages of our method are efficiency improved by not requiring additional human actions, and functionality to enable a robot to perform further actions. We consider that the efficiency is supported by users' physical workloads and cognitive loads. Users' physical workloads would not be increased because the proposed method does not require additional human actions. However, the effect of the proposed method on users' cognitive loads is unknown. We applied the method to a desktop sweeping task by a human and a small mobile robot, and conducted an experiment with participants to measure users' cognitive loads in a cooperative sweeping task. As a result, we found that the proposed method had a lower cognitive load than a typical conventional method.

Keywords interaction design · cooperative sweeping · cognitive load · mobile robot

K. Kobayashi
Graduate School of Science and Technology, Shinshu University, 4-17-1 Wakasato, Nagano City, 380-8553, Japan
E-mail: kby@shinshu-u.ac.jp

S. Yamada
National Institute of Informatics / The Graduate University for Advanced Studies (SOKENDAI),
2-1-2 Hitotsubashi, Chiyoda, Tokyo, 101-8430, Japan
E-mail: seiji@nii.ac.jp

1 Introduction

There has recently been an increase in research on robots for home use [1]. For example, an autonomous lawn mower called Robomow[1] and an autonomous sweeping robot called Roomba[2] have been developed for practical use. Thus, robots have moved from doing activities only in laboratories and industrial factories to home environments. However, in a home environment, users often face situations where they need to help the robots. For example, users need to move obstacles before they turn on the power of a sweeping robot to clean the entire floor of a room. This intervention makes a safe and easy environment for robots to work in. Although robots are introduced to home environment, human work will be always required. In addition, users need to learn the manipulation method and precisely manipulate them.

From the point of view of always requiring human work while using a robot, it will be appropriate to focus on human-robot cooperation. In the following studies about human-robot cooperation, human actions for achieving their task are regard as commands for a robot. In other words, users just do their job without robot manipulation. Hayashibara et al. [2] and Arai et al. [3] presented a robot that can cooperatively carry a long object with a human. Nakai et al. [4] presented the control method for mobile robots that carried heavy objects. Yokoyama et al. [5, 6] studied a cooperative carrying task performed by a human and a humanoid. In those studies, every robot senses the pushing or pulling force of an object by the user. Thus, the users can intuitively control the robots by force. However, this force-

[1] http://www.frietndlyrobotics.com/robomow/
[2] http://www.irobot.com/

based controlling method can only be applied to limited tasks where a user and a robot carry an object together.

In this paper, we propose a novel method to control a robot without robot manipulation like such object carrying tasks. Users do not need to precisely manipulate the robot and to learn the manipulation method. The proposed method can send commands to a robot by using a human action sequence that achieves their own task. In order to enable the robot to achieve tasks, we introduce a keep-based interaction in which a human keeps an action of the sequence for a certain period. In contrast with the above force-based controlling method, the proposed method is not limited to object carrying task. We applied the method to a desktop sweeping task. The advantages of our method are efficiency improved by not requiring additional human actions, and functionality to enable a robot to perform further actions.

We consider that such efficiency is supported by users' physical workloads and cognitive loads. Users' physical workloads would not be increased because the proposed method does not require additional human actions. However, the effect of the proposed method on users' cognitive loads is unknown. Thus we conduct an experiment with participants and measure users' cognitive loads in a cooperative sweeping task.

The rest of this paper is presented as follows. A classification of interaction models between a human and a robot is described in Section 2. The details for the proposed method based on one of the classified models are explained in Section 3. In Section 4, the proposed method is applied to a cooperative sweeping between a user and a small sweeping robot, and is compared to two conventional methods in terms of the cognitive loads of the users. We discuss the experimental results in Section 5 and describe the studies related to our work in Section 6. Finally, we conclude our study in Section 7.

2 Interaction Models

In this section, we classify some types of interactions between humans and robots into two groups, direct commanding methods (DCM) and commands embedded in actions (CEA). Based on the classification, we propose an extended model of CEA (ECEA).

2.1 Direct Commanding Method

Users sometimes directly control robots by using a remote control, voice commands, gesture command, etc. We call this type of method the *Direct Commanding Method* (DCM), in which users directly send commands to robots in order to control them. Plenty of studies have used DCM, such as gestures [7, 8, 9], speech recognition [10, 11, 12], and control devices like joysticks [13, 14, 15]. Figure 1 shows the DCM interaction model. In this model, a human has two tasks: to control the robot by sending commands and to arrange the environment. An example of an interaction between a sweeping robot and a human is described below, where H, R, and E represent the human, the robot, and the environment, respectively. In this example, the robot requires the user to arrange the environment in a way that the robots do not obtain complete autonomy.

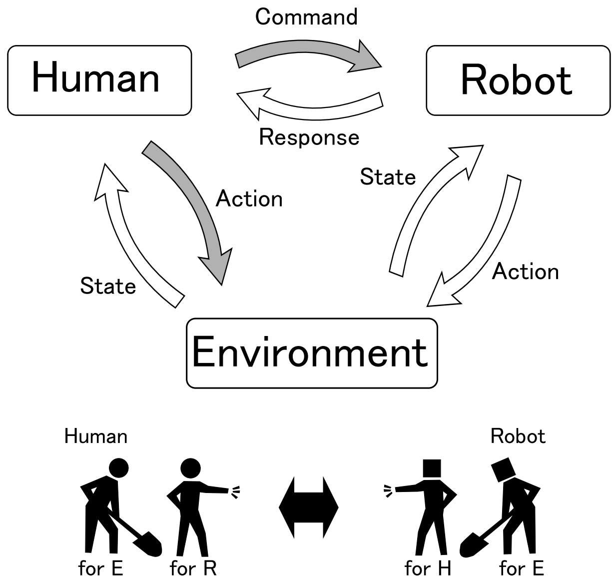

Fig. 1 DCM interaction model. A human directly sends commands to robots in order to control them. It includes methods using a remote control, voice commands, and gesture command.

- $H \Rightarrow R$: The human controls the robot using a remote control.
- $R \Rightarrow H$: The robot returns a response to the human about a command.
- $H \Rightarrow E$: The human removes obstacles in the environment.
- $E \Rightarrow H$: The state of the environment is recognized by the human.
- $R \Rightarrow E$: The robot sweeps the floor.
- $E \Rightarrow R$: The state of the environment is recognized by the robot.

We consider that there are actually many tasks that require actions from a user to an environment such as opening a door for a mobile robot, removing furniture for a security robot to monitor a room, pasting markers for a humanoid robot, and so on. Additionally, reading a manipulation manual, connecting cables to a robot, searching a remote control are also actions from a user to an environment because they are not commands for robots to perform a task.

2.2 Commands Embedded in Actions

The second category is named *Commands Embedded in Actions* (CEA), in which commands for robots are embedded in human actions. It has no direct command to robots. Users do not need to manipulate them. The purpose of human action by CEA is to play only a role to achieve a task without robot manipulation. In contrast, the purpose of human action by DCM is limited to sending commands. CEA enables users to act for themselves and robots to make a decision based on the user's action. Typical examples in the industrial field are automatic doors and faucets, e.g. a human approaches a door in order to go through it, and puts her/his hands under a faucet in order to wash the hands. In such situations, user's actions to only achieve her/his own task simultaneously become commands to a robot. Figure 2 shows a CEA interaction model. An example of an interaction between a human and an automatic door is described below.

- $H{\Rightarrow}E$: The human approaches the door.
- $E{\Rightarrow}H$: The state of the environment is recognized by the human.
- $R{\Rightarrow}E$: The robot opens or closes the door.
- $E{\Rightarrow}R$: The state of the environment is recognized by the robot.

In this interaction, there are no direct commands between the human and the robot, and the commands ($H{\Rightarrow}R$) and responses ($R{\Rightarrow}H$) are not configured. The human approaches the door because she/he wants to go through the door. She/he does not want to operate the robot.

It appears that CEA reduces the users' physical workloads because they do not need to perform additional actions and to learn particular methods to communicate with robots. In fact, we can easily find applications of CEA, such as for automatic doors and faucets, and they are certainly convenient in our daily lives although they do not have any complicated sensors.

The advantage of CEA is not to require additional human action, and the advantage of DCM is to enable a robot to perform further actions. In this study we propose a method that has the advantages of CEA and DCM. By extending CEA, it will be possible for a robot to cooperate with a user. Cooperation between a user and a robot will easily perform a task and achieve a task that is difficult for a user to perform by herself/himself because she/he can perform remaining tasks instead of commanding a robot. Our proposed method is appropriate for cooperative tasks because it provides time for a robot to work autonomously while a user performs a task. In the next section, the details of our proposed method are described.

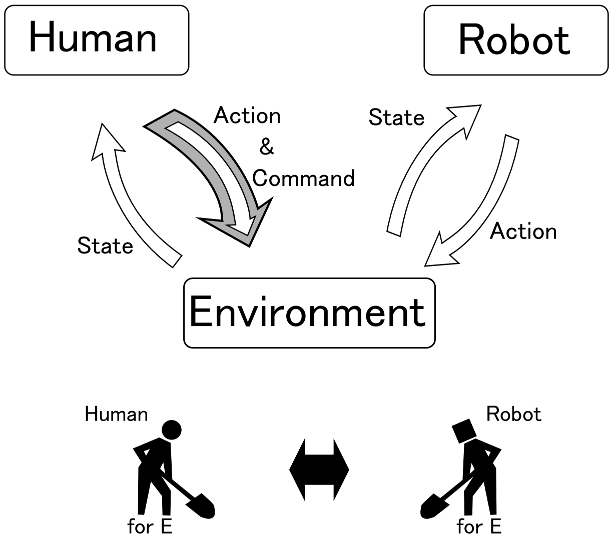

Fig. 2 CEA interaction model. Commands for robots are embedded in human actions. It has no direct command for robots. Users do not need to manipulate them and enables users to act for themselves and robots to make a decision based on the user's action.

3 Extension of Commands Embedded in Actions

Robots can be conveniently controlled by CEA because a user does not need to learn the control methods of the robot. However, CEA is limited to enable a robot to perform an action. We then extend CEA to enable the robot to perform further actions without additional human actions. We call the proposed method *Extended CEA* (ECEA). The advantages of ECEA are efficiency improved by not requiring additional human actions like CEA, and functionality to enable a robot to perform further actions like DCM. ECEA requires less human actions than DCM, and enables a robot to perform more actions than CEA. ECEA is appropriate for cooperative tasks because it provides time for a robot to work autonomously while a user performs a task. In our study, we apply ECEA to human-robot cooperative tasks.

3.1 Description of human action

It is an important factor to how to divide and describe human behavior when designing human-robot interaction. There have been some studies on the description methods of human behavior in the psychology and cognitive science fields. The action coding system (ACS) [16] was developed to analyze the everyday behavior of brain-damaged patients. It segments human behavior into a sequence of primitive actions using a criterion of whether the action includes a change of an object's position or state in an environment. This kind of approach has been used in various studies analyzing human behavior. ACS is one of the most promising ways in our study to segment a human action.

Another similar work was conducted by Newtson et al. [17, 18, 19]. In their experiments, a movie of an actor's behavior was presented to the participants and they were requested to push a button when the participants recognized the division of natural and meaningful primitive actions. The results showed that they found that each participant pointed out almost the same division of primitive actions. They called this division a *break point.* Furthermore, they investigated how easily the participants understood the behaviors from several snapshots taken from the movie. The participants were requested to answer by showing the correct order sequence of the snapshots. Half of the participants were given several snapshots including break points, and the other half were given ones not including break points. The results showed that more participants given the snapshots with the break points answered correctly than the other half even though the same number of snapshots was given to both groups.

We use the segmentation by ACS because it is adequate to decide time points to insert a new action. Newtson's experiment showed that such segmentation is understandable to observers. There is no guarantee that such segmentation is also understandable to actors. However, we consider that it is helpful for users to perform a new action. The details of ACS are described below.

3.2 Action coding system

The action coding system deals with the two levels of actions described below.

– *A-1 unit:* This unit is the smallest components of a behavioral sequence that achieves a concrete, functional result or transformation, describable as the movement of an object from one place to another or as a change in the state of an object.

(1) MOVE (x) TO (location) VIA (instrument) BY (manner)
(2) ALTER (x) TO (location) VIA (instrument) BY (manner)
(3) TAKE (x) (i.e. take possession of object x)
(4) GIVE (x) (i.e. relinquish possession of x)

Fig. 3 Four basic A-1 units of ACS description.

Opening sugar pack and pouring sugar:

TAKE sugar pack
ALTER sugar pack TO open BY tearing
MOVE sugar TO in coffee VIA pack BY pouring

Fig. 4 ACS description of opening sugar pack and pouring sugar.

– *A-2 unit:* This unit is a group of A-1 units that accomplishes one of the basic sub goals of a task. In our study, we do not use A-2 units because we focus on a primitive action that users can easily understand and execute.

Schwartz et al. [16] coded the four A-1 units shown in Fig. 3 by observing people who made breakfast. For example, the action of *opening sugar pack* and *pouring sugar into coffee* are described in Fig. 4. ACS has simple and repeatable description policies and uses two description guidelines: (1) describe general and recognizable things (open/close, on/off) and (2) select subjects that are most related to the task. ACS description can divide user's actions in parallel. For example, left hand actions and right hand actions are described in parallel. In our study, we assume the sequential ACS description.

3.3 Introducing a human action *KEEP*

The advantage of using CEA is that users can perform unconscious behaviors and have less of a physical workload. An introduced action should be easily executed by a user. We introduce a *KEEP* action as an A-1 unit based on ACS. The *KEEP* is defined as *an A-1 unit that keeps the last state of the previous unit.*

For example, the action of *opening sugar pack and pouring sugar* described in Fig. 4 are extended in Fig. 5 with *KEEPs*. In Fig. 5, two *KEEP*s are added, which represent the *keeping the last state of taking* and *keeping the last state of altering*, respectively. A user performs the *KEEP*s by keeping the given body posture. This duration of keeping depends on the tasks and the user's decision. The introduced action *KEEP* is easily performed by users because they only have to maintain their posture. There is no new action and no need to

Opening sugar pack and pouring sugar:

TAKE sugar pack
KEEP TAKE (keep last state of TAKE)
ALTER sugar pack TO open BY tearing
KEEP ALTER (keep last state of ALTER)
MOVE sugar TO in coffee VIA pack BY pouring

Fig. 5 ACS description opening sugar pack and pouring sugar using *KEEP* action.

learn anything. A robot detects the posture as a command and performs actions according to the human. The action *KEEP* should not be applied for the action that continues to change the state of objects. For example, it is not appropriate to apply *KEEP* to the pouring action (MOVE sugar TO in coffee VIA pack BY pouring) in Fig. 5 because the action continues to pour the sugar too much.

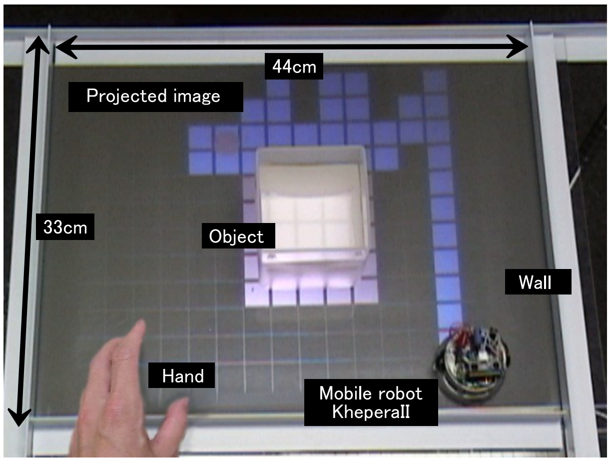

Fig. 6 Sweeping area. It has a flat surface to be swept, a wall enclosing the sweeping area, and a box placed in the center of it. Based on the robot's location, the projected image includes small square cells and it is immediately colored when the robot enters the cell. The colored cells represent the region swept by the robot.

4 Experiment

We design robot behavior to achieve a cooperative sweeping between a human and a robot. In the sweeping, the human helped the autonomous sweeping robot by removing an obstacle. We actually verified the realization of a cooperation using ECEA. Then, we conducted an experiment to investigate the participants' cognitive load when they cooperate with a robot using ECEA in comparison with typical DCM.

4.1 Design and execution of cooperative sweeping

We achieved a cooperative sweeping between a human and a small mobile robot in a desktop-like environment. The goal of the task was to sweep off the top of a desk including the area under an object placed on the desk. Details about environment, task, specifications of the small mobile robot with behavioral rules, and the method are described below.

4.1.1 Environment and cooperative task

We used the desktop-like environment shown in Fig. 6, which had a flat surface to be swept (44 × 33 cm), a wall (height: 1.5 cm) enclosing the sweeping area, and a box (dimension: 7 × 7 × 7 cm) placed in the center of it. The box simply simulated an object likely being on a desk, such as a TV remote, a mobile phone, or a pen holder.

In this environment, we defined the goal of the cooperative task as *to sweep off the desk including the region under the object*. A human and the robot work together to achieve the goal.

4.1.2 Mobile robot

We used a small mobile robot called KheperaII (Fig. 7). The robot had eight infrared proximity and ambient light sensors with up to a 10 cm range (1–8 in Fig. 7), a Motorola 68331 (25MHz) processor, 512 Kbytes RAM, 512 Kbytes Flash ROM, an two DC brushed servo motors with incremental encoders. A program written in C-language was run on the RAM. In Fig. 7, the sensors 1–6 and 8 directed to side, and the sensor 7 is directed upward. We rearranged the sensor 7 and added the sensor 9 (SHARP GP2D12) with up to an 80 cm range for sensing a lifted object.

The robot performed the reactive behavior using the local information gather from the sensors equipped on it. It had no model of the environment or learning mechanism. The reactive behavior can be simplified for the implementation of the robot. We thought that the robot had sufficient enough specifications for the task and its low cost provides an advantage in a realistic situation. In fact, the specifications were similar to those of commercial sweeping robots such as Roomba.

4.1.3 Interaction design for cooperative work

The cooperative work scenario between a human and a robot is described as follows. First, when a human picks up the object and keeps it held over the desk, the robot goes to the region under the object and sweeps it out. Next, when the human puts the object down, the robot goes other places and autonomously sweeps the

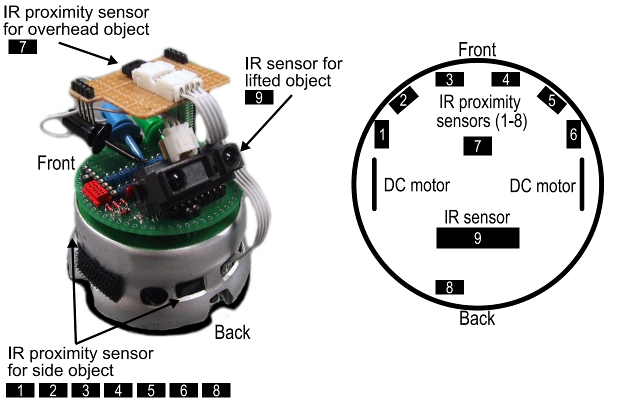

Fig. 7 KheperaII robot. The robot has eight infrared proximity and ambient light sensors with up to a 10 cm range (1–8). The sensors 1–6 and 8 directed to side, and the sensor 7 is directed upward. The sensor 7 is rearranged and the sensor 9 (SHARP GP2D12) with up to an 80 cm range for sensing a lifted object are added.

desk. This cooperative work achieves the defined goal. We call this cooperative work "interactive sweeping".

To perform interactive sweeping, we assign tasks to the human and the robot. The robot's task is *to sweep the desk* and the human's task is *to move the object*. The human and robot tasks achieve *sweeping off the desk including the region under the object*. It is important that the task assigned to the human is achieved by performing usual and unconscious actions that would be a standard part of a sweeping task performed by two humans. The human does not need to perform unusual actions that she/he has never performed before, such as the manipulation of a robot or using the remote control for a machine.

The robot performs a random sweeping that is to repeat turning to a random direction and then going straight. Since the robot has no cleaning mechanism, we assume that the region under the robot is cleaned. The robot does not use effective region covering methods for sweeping [20], because those would need the robot's exact location which would be difficult to obtain. For example, a dead reckoning method is not very reliable because of its accumulated errors.

4.1.4 Applying ECEA and CEA

By applying ECEA and CEA to the tasks of the human and the robot, they are able to perform the interactive sweeping described above. The robot quickly moves to the area under the object when the human moves it. First, we segment the human behavior into a sequence of primitive actions. Figure 8 shows the four primitive human actions of moving the object based on ACS. The typical human behaviors where they grasp the object

(1) TAKE object	
(2) MOVE object TO z	(z: a vertical location)
(3) MOVE object TO x	(x: an initial object's location)
(4) GIVE object	

Fig. 8 ACS description of typical human actions to move object.

(TAKE object), raise it (MOVE object TO z), lower it (MOVE object TO x), and releases it (GIVE object). Although there are different A-1 unit combinations to achieve the human task, we use these four actions because they seem to be the simplest and most rational movements in the sweeping task.

To achieve interactive sweeping, we synchronize the robot's actions with the human's actions. Considering the human actions shown in Fig. 8, we make the robot perform the following three behaviors in sequence.

`Approach the object.`	(B1)
`Sweep the area under the object.`	(B2)
`Go out of the area under the object.`	(B3)

As shown in Fig. 9, ECEA is applied to B2, and CEA is applied to B1 and B3. The robot approaches the object (B1) when the human raise it (Fig. 9(2)), and it goes out of the area when she/he lowers it (Fig. 9(3)). In order to achieve B2, we then add an A-1 unit *KEEP*, which keeps the last state of the previous action, between the A-1 units (2) and (3) in Fig. 8. The *KEEP* is where the human maintains the last state of movement, raising the object. The robot sweeps the area under the object while a *KEEP* is performed. The robot continues to sweep the area under the object while it is lifted. The human observes the sweeping progress made by the robot, and moves it back into its original position when she/he thinks that the area is clean.

4.1.5 Behavioral design of mobile robot

We design the robot to achieve random and interactive sweeping. We additionally attach the two range finders shown in Fig. 7 to sense the lifted object and to measure the distance of the object placed over it. The sensor for sensing the lifted object faces 45° upwards. The robot spins periodically to find the lifted object in the random sweeping mode.

We use the behavior-based approach and the subsumption architecture [21] to implement the robot behavior. The actions in the layers of the subsumption architecture are asynchronously performed. The higher layers suppress the lower ones including the actions that are more reactive then those in the upper layers. Each

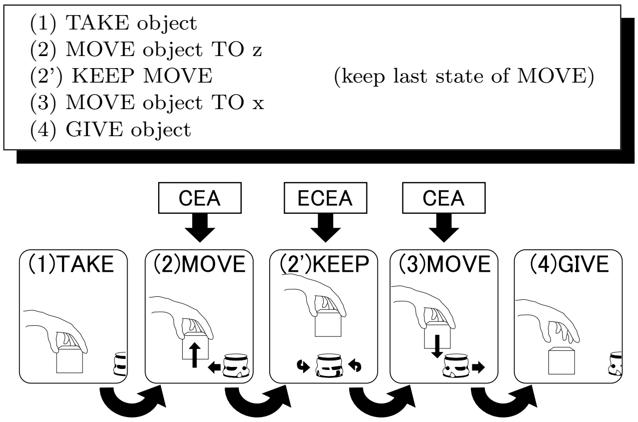

Fig. 9 ACS description of moving object using *KEEP* action. ECEA is used in (2'), and CEA is used in (2) and (3). The robot approaches the object when a user raises it (2). It sweeps the area under the object while the object is held above the area (2') and goes out of the area when the user lowers it (3).

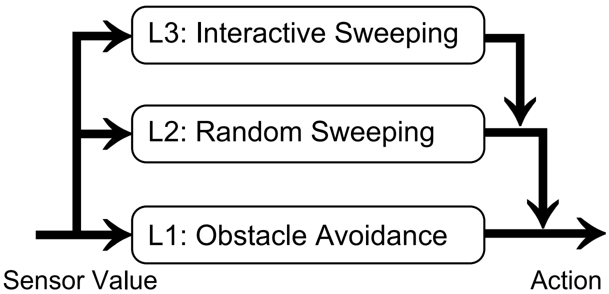

Fig. 10 Subsumption architecture. We designed three layers of robot behavior; the obstacle avoidance layer, the random sweeping layer, and interactive sweeping layer.

layer consists of multiple actions and selects an action to perform according to the sensor value. When there are multiple candidates from several layers, only one action is selected as the output action. Figure 10 shows the robot's behavior in the three layers based on the subsumption architecture. Each layer had its own intervals to output an action. We set the intervals at five msec for the obstacle avoidance layer, ten msec for the random sweeping layer, and five msec for the interactive sweeping layer.

The following lists are the behavior rules in each layer. The rules are described in IF-THEN forms in which the precondition part contains the robot's state based on the sensor responses and the action part contains the motor command.

Layer 1: Obstacle avoidance

This layer deals with the most primitive behavior. The robot mainly performs a '`stop`' action. The rules and actions in this layer are listed below. The robot avoids a collision when it is moving forward (L1-01) and backwards (L1-02). In the precondition parts, '`in the front`' represents a response from sensors 3 or 4 and '`in the rear`' represents a response from sensor 8 in Fig. 7.

```
IF robot is moving forward ∧            (L1-01)
object is in front THEN stop
IF robot is moving backward ∧           (L1-02)
object is in rear THEN stop
```

Layer 2: Random sweeping

This layer makes the robot sweep the desk and search for the lifted object. The main actions of the robot are '`move forward`' and '`spin`.' The rules and actions in this layer are listed below. The robot rotates itself to search for the lifted object (L2-01) and spins itself around when it senses the 'unlifted' object or a wall at its left (L2-02) or right front (L2-03). The robot performs the 720 degrees spin because we empirically find it is faster to detecting the lifted object than 360 degrees. In the preconditions, '`in the left front`' represents a response from sensors 2 or 3 and '`in the right front`' represents a response from sensors 4 or 5.

```
IF no object is in the front THEN       (L2-01)
move forward, rotate 720°, and
move forward again
IF object is in the left front          (L2-02)
THEN spin clockwise at random
from 90° to 180°
IF object is in the right front         (L2-03)
THEN spin counter-clockwise at
random from 90° to 180°
```

Layer 3: Interactive sweeping

This layer achieves interactive sweeping. The robot approaches the lifted object (L3-01), sweeps the area under the object according to the human action (L3-02), and goes out of the area when the object comes close to its top (L3-03). In the preconditions, '`lifted object is sensed`' represents a response from sensor 9, '`going out of the area under object`' represents the sensor 7 value being reduced, and '`object is on its top`' represents the sensor 7 value representing a distance of less than 0.5 cm. The decrease in sensor value indicates increase of distance. When several conditions are met, only one action is randomly selected.

IF lifted object is sensed THEN move forward (L3-01)
IF it is going out of the area under object THEN go backward, spin clockwise at random from 90° to 180° (L3-02)
IF object is close to its top THEN move forward (L3-03)

4.1.6 Execution of cooperative sweeping

We confirmed that the robot achieved its goal of the assigned task in which it swept out the desk. Figure 11 shows some photographs of the cooperative sweeping by the robot and a human. These photographs were extracted from a movie every one second. In Fig. 11(1) to (5), the robot autonomously sweeps the top of the desk using a random spinning strategy and searches for the lifted object. The robot finds the lifted object (6) and approaches it (7). In photographs (8–10), it sweeps the area under the object. By applying CEA and ECEA, the robot and the human achieved the cooperative task. Thus, we confirmed the achievement of a cooperative sweeping by ECEA as one of the practical cooperative tasks.

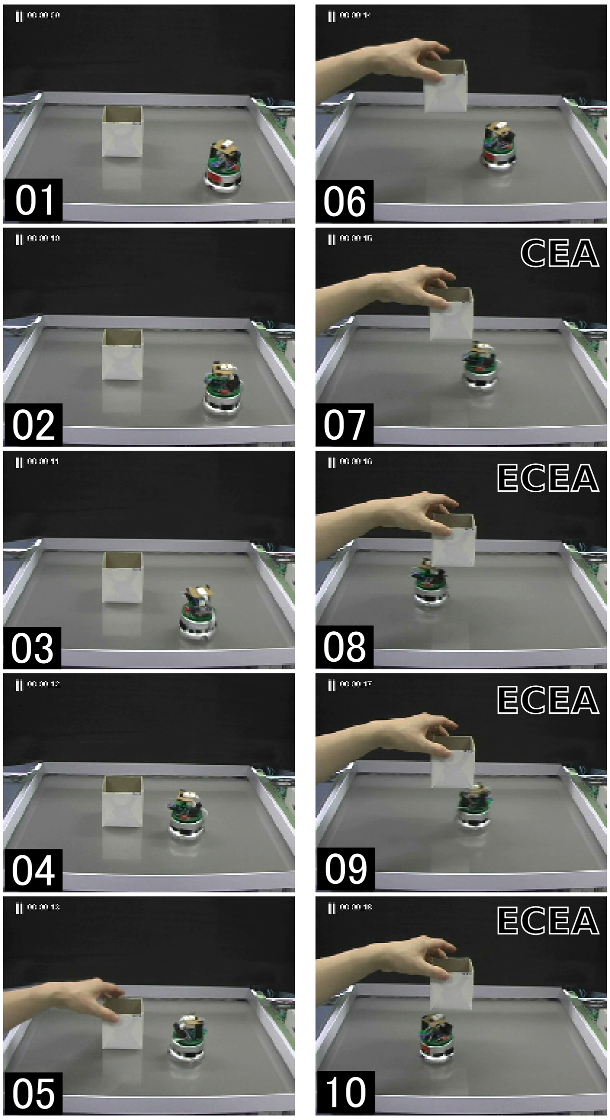

Fig. 11 Execution of cooperative sweeping between a human and a robot. The robot autonomously sweeps the top of the desk using a random spinning strategy and searches for the lifted object (1–5). It finds the lifted object (6) and approaches it (7), and sweeps the area under the object (8–10).

4.2 Experiment: Measuring cognitive load

We investigated the user cognitive load while they controlled the robot. A comparison of the control methods between ECEA and DCM was performed by uniforming the robots' specifications other than control methods. We describe the ECEA characteristics using the results from this experiment.

Figure 12 shows the human actions in the sweeping tasks by ECEA and DCM. ECEA enables a user and a robot to achieve the sweeping task cooperatively. In contrast, DCM dose not promote cooperation between a user and a robot. Both these method have the same purpose, to achieve the sweeping task. There is less number of human actions in ECEA than in DCM because a human with DCM has to move the object to another place before sending commands. However, cognitive load for a human while controlling a robot is unknown. In this case, the targets for comparison are the third ECEA action and the fourth DCM action in Fig. 12. We investigated the cognitive load of users while controlling the robot to sweep under the object. The cognitive load for ECEA and DCM were compared.

4.2.1 Experimental system

We develop an experimental system to make it easy for a user to interact with a robot. Figure 13 shows the experimental system displaying the area swept by the robot. This system consists of the sweeping area shown in Fig. 6 and the devices under the sweeping area including a laptop computer, a projector, and a USB camera with an infrared filter. The sweeping area is the same as in section 4.1.1 except that the swept area is colored. The devices under the sweeping area create an image to indicate the swept area and project it on the desktop. In the projection process, the robot's

	ECEA		DCM (typical)
1	TAKE object	1	TAKE object
2	MOVE object TO z	2	MOVE object TO y
3	KEEP MOVE	3	GIVE object
4	MOVE object TO x	4	SEND command
5	GIVE object	5	TAKE object
		6	MOVE object TO x
		7	GIVE object
			(y: a object's location)

Fig. 12 Human actions in the sweeping tasks in ECEA and DCM. There is less number of human actions in ECEA than in DCM because a human with DCM has to move the object to another place before sending commands.

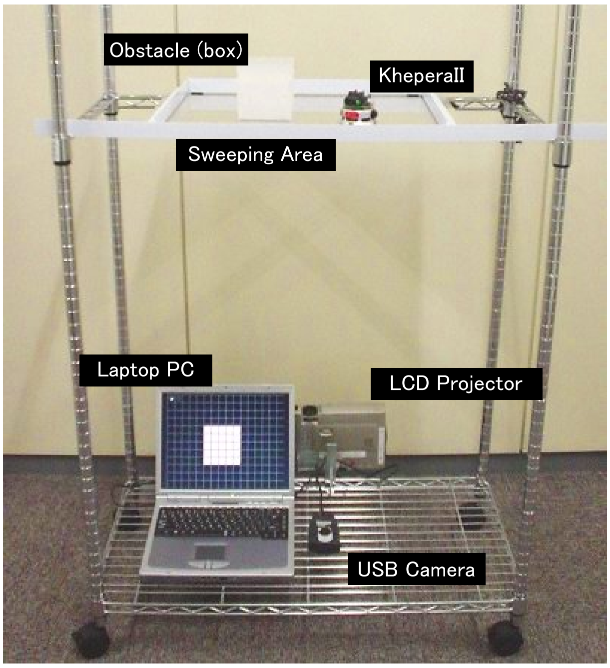

Fig. 13 Experimental system. The devices under the sweeping area include a laptop computer, a projector, and a USB camera with an infrared filter. The robot's location is obtained from the camera image that includes two infrared LED beams equipped on the robot. Based on the robot's location, it creates a projection image that includes small square cells and it is immediately colored when the robot enters the cell.

location was calculated by processing the picture from the camera. The robot's location was obtained from the camera image that includes two infrared LED beams equipped on the robot. Based on the robot's location, a projection image is created. The projected image includes small square cells and it is immediately colored when the robot enters the cell. The colored cells represent the region swept by the robot. The desktop was 44 cm wide and 33 cm long and was divided into 16 × 12 cells.

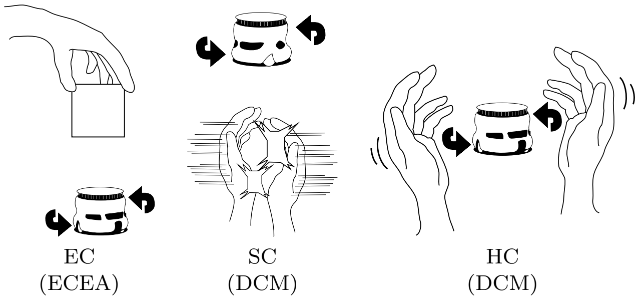

Fig. 14 Three types of interactions in the experiment. In EC condition, the robot is controlled by ECEA. The users pick up the object and hold it above the area on which they requested to be swept. In SC condition, the robot is controlled by DCM. The users clap their hands to make a sound just before the robot goes out of the area on which they requested to be swept. In HC condition, the robot is also controlled by DCM. The users move their hands in front of the robot just before it goes out of the area on which they requested to be swept.

4.2.2 Mobile robots

We used three kinds of robots that were controlled in different methods. One of the three robots was controlled by ECEA and the other two were controlled by DCM. The robot controlled by ECEA had the same behavioral rules as that in section 4.1.5. In order for users to easily control the robot with DCM, we introduced a controlling method by using a single command without its own controllers. The two methods were chosen as the typical DCMs needing no remote control devices. These control methods are shown in Fig. 14, and the details are described as follows:

EC (ECEA): The robot is controlled by ECEA; it spins at random from 90° to 180° when it senses the edge of the object placed over it. The users pick up the object and hold it above the area on which they requested to be swept.

SC (DCM): The robot is controlled by DCM; it spins at random from 90° to 180° when it senses a sound. The users clap their hands to make a sound just before the robot goes out of the area on which they requested to be swept.

HC (DCM): The robot is controlled by DCM; it spins at random from 90° to 180° when it senses an obstacle in front of it. The users move their hands in front of the robot just before it goes out of the area on which they requested to be swept.

In these DCMs, the commands for the robots are not embedded in human actions. The robots receive these commands by sensing using extra sensors, such as microphones or a slight program modification of the robot. We implemented SC and the HC by changing the precondition of L3-02 in the behavior rules from '`it is going out of the area under object`' described in the section 4.1.5 to '`sound is detected`' and '`object is in the front`', respectively.

4.2.3 Method

We measured the cognitive load of the participants interacting with the three robots. The measurements started when the robot entered the region they were required to sweep, and it continued until all the cells of the region were swept. The object was placed in the center of the sweeping area. Under the EC condition, the participants kept picking up the object until the area under it was completely swept. Under the SC and HC conditions, the participants first replaced the object in a corner of the sweeping area and then controlled the robot by clapping and moving their hands to completely sweep the region.

We used a dual task method to measure the participants' cognitive loads. The participants had to do mental arithmetic as a secondary task [13, 22]. The primary task was to control the robot. They vocally counted backwards by three from a three-digit number selected at random. We obtained the number of correct answers per second and evaluated them as the participants' cognitive load for controlling the robots. Our hypothesis of the experiment is that participants provide more correct answers in the EC condition than that in other conditions.

4.2.4 Instructions and work flow

We showed the following text to the participants and read it out as instructions.

Introduction: The purpose of this experiment is to investigate the usefulness of robots. We take video pictures of your trials and analyze it after this experiment. We do not use the video data and personal information except for the investigate purpose. If the video is used for a conference presentation, we will contact you again. Thank you for your cooperation.

Abstract: You work together with a robot to clean the top of the table. The cleaned area glows instead of vacuuming off because the robot does not have the function of vacuuming off. There is a box on the table. Your task is to control the robot properly and make it to clean the area under the box. The area under the box glows red. When the robot goes through the area, the color is changed from red to green. Please change the all of red area to green by controlling the robot.

In addition, you perform another task with controlling the robot at the same time. The task is to vocally count backward by three from a three-digit number. For example, when the number 231 is provided, you vocally count it backward like 231, 228, and 225 while controlling the robot.

Precautions:
- When you count the number, please speak loudly and clearly.
- Do not pick up the robot and push it strongly.

Procedure:
- Three conditions are provided; condition A, condition B, and condition C.
- Each condition provides a different control method of the robot.
- For example, control the robot by making sound and moving your hand in front of it to change its direction.
- In a condition, you perform a practice and a measurement. We do not take video pictures while in the practice.
- The purpose of the practice is to sufficiently get used to the control method.
- The procedure in each condition is the same.
- A condition is provided by the experimenter.

1. Experimental system:
   - You control the robot on the translucence table.
   - There is a grid on the table. The squares the robot goes through are colored.
   - The red colored squares are the area needed to be cleaned. They are changed to green when the robot goes through there. You need to control the robot and change all of their color green.
   - The red circle at the right corner on the table is the start point of the robot.
   - There is a box and you need to manipulate it as the experimenter explained later.
2. Preparation:
   - When the experimenter provides a start signal, numbers are displayed at the right corner on the table like 3, 2, 1, START. After displaying "START", a three-digit number is displayed. This is the first number you need to count backward.
   - When the three-digit number is displayed, please say it.
   - After that, say the subtracted number by 3 continuously. When you count the number, please speak loudly and clearly as possible as you can.
   - If you notice a mistake in calculation, continuously use the mistaken number.

- Please prioritize the control of the robot over the calculation.
3. Practice: (Until controlling the robot without mistakes. For five minutes.)
   - The experiment begins in the condition X. Please refer the condition X in the next section of "Control manual of the robot". (Actually a condition code is written in X.)
   - After displaying "START" on the table and you count the number three times, the experimenter begins to move the robot. When the robot begins to move, you interact them according to it.
   - Manipulate the box properly as described in the control manual.
   - Watch the experimenter's demonstration.
   - Practice the robot control several times.
   - When you get used to control the robot, gradually speed up your number counting.
   - Also, you practice the number counting few times without controlling the robot until you smoothly perform it.
4. Measure:
   (1) Execution of mental arithmetic for 30 seconds without controlling the robot.
   (2) Execution of three trials with video recording.
5. go to 3 and begin practice in another condition.

Control manual of the robot

- Condition A
  - In the condition A, you can change the robot's direction by using the box.
  - At the beginning of the starting signal, you pick up the box.
  - When the robot enters the area under the box, it sweeps out there by repeating direction change.
  - It spins at random from 90° to 180° when it senses the edge of the object placed over it.
  - Practice fixing the box above the area to be swept. It will be better to keep as still as possible.
  - It is failure that the robot goes out of the area under the box and does not come back soon.
- Condition B
  - In the condition B, you can change the robot's direction by sound.
  - At the beginning of the starting signal, you replace the box in a corner of the sweeping area.
  - Clap your hands to control the robot. It will be better to clap your hands a little harder.
  - It spins at random from 90° to 180° when it senses a sound.
  - It is failure that the robot does not spin when you clap your hands or it goes out of the area to be swept far away.
- Condition C
  - In the condition C, you can change the robot's direction by using your hands.
  - At the beginning of the starting signal, you replace the box in a corner of the sweeping area.
  - The robot finds your hands in front of it, it changes its direction. It will be better to make a fence by your hands around the area to be swept.
  - It spins at random from 90° to 180° when it senses your hands in front of it.
  - It is no problem to slightly touch the robot.
  - It is failure that the robot goes out of the area to be swept far away.

Table 1 Participants work flow of the experiment.

No.	Event
1	A condition is provided
2	Practice for 5 minutes
3	Mental arithmetic without the robot
4	Trial 1
5	Trial 2
6	Trial 3
7	Another condition is provided
8	Practice for 5 minutes
10	Mental arithmetic without the robot
11	Trial 1
12	Trial 2
13	Trial 3
14	The last condition is provided
15	Practice for 5 minutes
16	Mental arithmetic without the robot
17	Trial 1
18	Trial 2
19	Trial 3

Table 1 shows the participants work flow of the experiment and photographs of the experiment are shown in Fig. 15.

4.2.5 Results

Eight men and four women between the ages of 22 and 32 participated in this experiment. Figure 16 shows the participants' averaged scores and standard deviations. The scores were normalized by each participant's calculation ability. A score of 1.0 represents their calculation ability without controlling the robot. The Dunnett's multiple comparison test was used to statistically identify the significant differences in the scores from the control group (EC). The difference between EC–HC was significant ($t = 3.938, p < 0.01$) and EC–SC was also significant ($t = 2.414, p < 0.05$).

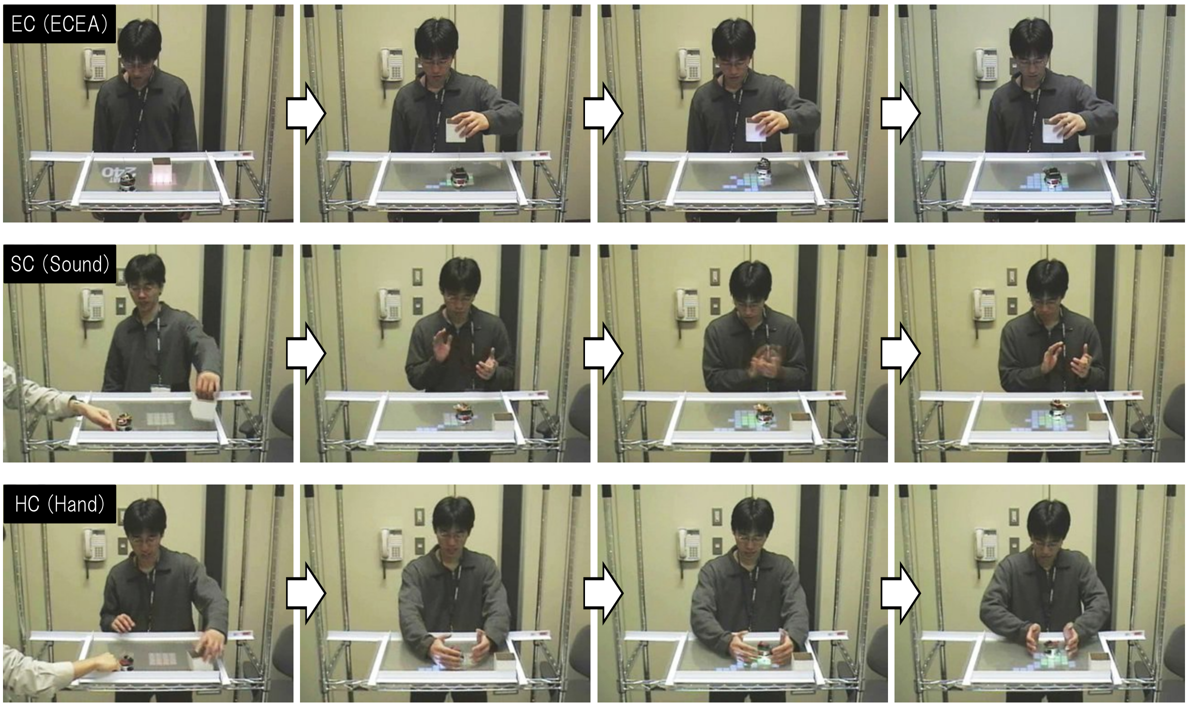

Fig. 15 Photographs of the experiment. The top image sequence is the EC condition. The user picks up the object and holds it above the area on which they requested to be swept. The middle image sequence is the SC condition. The user claps their hands to make a sound just before the robot goes out of the area on which they requested to be swept. The bottom image sequence is the HC condition. The user moves their hands in front of the robot just before it goes out of the area on which they requested to be swept.

The average sweeping times of EC, SC, and HC were 20.89, 21.46, and 18.25 seconds, respectively. The Dunnett's test was also used to statistically identify the significant differences in sweeping times from the control group (EC). There was no significant difference between EC–SC ($t = 0.260, p = 0.952$) and EC–HC ($t = 1.203, p = 0.380$). The result of the experiment suggested that ECEA requires a lower cognitive load than DCM. In addition, it would appear that sweeping time of EC is not largely different from that of SC and HC.

5 Discussion

5.1 Cognitive load of ECEA

Even though there was the possibility to increase cognitive loads of users by using ECEA, the experimental results suggest that ECEA does not require a lot of cognitive load for the users. The *KEEP* action was easily performed by users. We believe that ECEA provides a practical method for achieving cooperative work between a human and a robot because tasks that the robot cannot perform on their own are achieved with a little help from the users.

In the experiment, since the robot performs random rotating actions, users cannot predict the direction of the robot. This unpredictability might lead the increase of users' cognitive load. However, we consider that it is more appropriate to use random rotating action than to use a joystick to control the robot. Using a joystick requires a selection of viewpoints such as a first-person viewpoint and a third-person viewpoint. In a first-person viewpoint, a user controls a robot in a relative coordinate to the robot as she/he rides on it. In a third-person viewpoint, a user controls a robot in an absolute coordinate. We considered that there are large differences in preference for viewpoints among users with using a joystick, and it is difficult to prepare an adequate viewpoint to each user in advance. Thus, we used DCM methods with less uncertainty than using a joystick. In addition, constant angle rotation could be available too. In this case, the user will eager to control the robot to go to the area not swept. It is also difficult to control the robot to certain direction. In the meantime, we removed the factors that increase users' cognitive load in our experiment. Actually, the participants of our experiment practiced the robot control before the measurement and there was no problem.

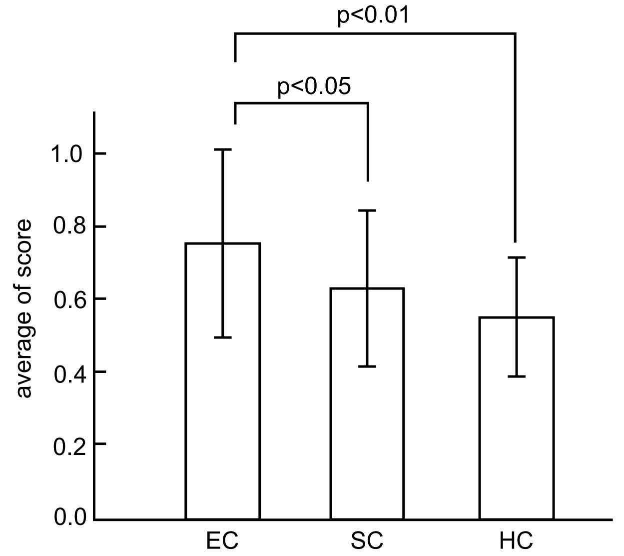

Fig. 16 Results of scores and differences. The scores are normalized by each participant's calculation ability. A score of 1.0 represents their calculation ability without controlling the robot. The Dunnett's multiple comparison test was used to statistically identify the significant differences in the scores from the control group (EC).

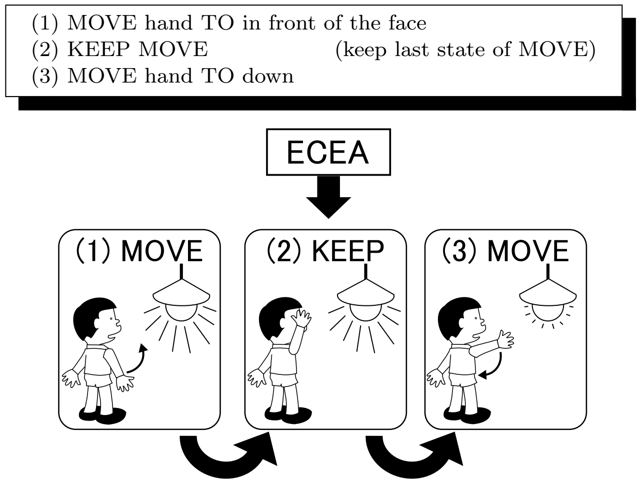

Fig. 17 ACS description with ECEA of room light dimming. ECEA is used in (2). Users send the command to dim the lights by screening their eyes from it with the hand.

5.2 Coverage of ECEA

In this section, we introduce three representative scenarios: *control of the light by actions*, *a garbage collection with a robot*, and *a cooperative cooking*. These examples imply generality of our proposed ECEA method. If the action *KEEP* is not applied for the action that continues to change the state of objects, our proposed method will be applicable to some cooperative tasks.

The proposed method has a possibility of applying it not only to robots but also to ordinary machines such as home electric appliances. Let's imagine a system that controls the lighting in a room. Users send the command to dim the lights by screening their eyes from it with the hand. This type of action unconsciously occurs when users think that the light is strong. They just hold this posture and then the system dims the light. When users want to brighten the light, they keep gazing at it until the light becomes a preferable strength. The user does not always need to keep gazing at the light because it is dazzling. Actually, the function of increasing the light intensity is achieved by turning the user's face toward the light. This type of system might be useful for them because they can determine whether they are satisfied or not with this action and reaction after they have performed it. The ACS description with ECEA of room light dimming is shown in Fig. 17. The *KEEP* unit that keeps the last state of MOVE (holding the hand in front of the face) is added. It will be developed by simple image processing with an infrared sensor. CEA is not able to achieve this kind of tasks because it is suitable to achieve tasks with only on/off control. In contrast, DCM can achieve this kind of tasks. However, in this case, a user needs to repeat a command such as saying "Darker" or to look for the remote control.

In addition, the proposed method is applicable to a garbage collection task. Robots are generally difficult to distinguish whether an object is garbage or not. So, it will convenient for a user that a robot comes close to her/him and takes the garbage. The user does not need to go to a trash can placed far away and just does pretend to throw the garbage to a trash can. The ACS description with *KEEP* of garbage throwing to the trash can is shown in Fig.18. The *KEEP* unit that keeps the last state of searching for the trash can (facing the trash can) is added into the original behavior. When the robot detects the action by using its infrared sensors or image processing, such as calculating the optical flows, it approaches the user. Thus, with the application of the ECEA, the robot might easily distinguish whether the user has garbage or not, because the operation *KEEP* in ECEA gives a good guidance for the robot to approach the user by showing stop motion and the garbage. CEA is not able to achieve this kind of tasks. The robot will reacts every user's action including throwing the garbage by using CEA. In contrast with DCM, ECEA does not need to look for a remote control or to send a command by saying "Come here" in a loud voice in a noisy environment.

There is another example in domestic tasks. ECEA is applicable to a cooking task. Although it is technically difficult for a robot to cook a meal from beginning to end by itself, it can help a user who cooks a meal.

(1) TAKE object
(2) MOVE object TO trash can
(3) KEEP MOVE (keep last state of MOVE)
(4) MOVE object TO robot

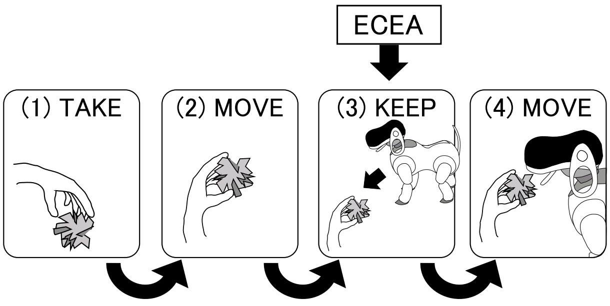

Fig. 18 ACS description with ECEA of garbage throwing. ECEA is used in (3). When the user pretends to throw the garbage to the trash can in (3), the robot approaches the user and take the garbage.

The ACS description with *KEEP* of a cooking task is shown in Fig. 19. A *KEEP* unit that keeps the last state of moving to reach the hand out for a spices bottle is added into the original behavior. When a robot detects such an action, it picks up the spice bottle and approaches to the user to pass it to her/him. Another *KEEP* unit keeps the last state of moving a frying pan to place food in a serving dish. When the robot detects such an action, it gives a dish to a user. Although the robot actually needs to manage contexts of cooking, it would play as a cooking helper. CEA is not able to achieve this kind of tasks because it makes a robot frequently react to every user's action. In contrast with DCM, ECEA does not need to push a button on a robot or a remote control, and to send a command by saying "Bring the spice bottle here."

5.3 Limitations

Although the experimental results show the availability of ECEA, it has some limitations. In the cooperative sweeping, proposed method is useful to clean the area under a lightweight and small object. It is not realistic to lift a large and heavy object such as a sofa.

If a robot can detect a region to sweep out on its own, the user could give a single command. It will be possible to apply the laser pointer interface to pick up an object on the floor [29]. However, it is technically difficult to develop such a robot as seeing from the previous work [29]. In our experiment, the robot randomly changed its direction for every experimental condition. It appears to be technically reasonable to achieve a sweeping task. ECEA does not always provide the best solution, but it may be able to reduce the technical difficulties to achieve a task.

(1) MOVE hand TO spices
(2) KEEP MOVE (keep last state of MOVE)
(3) MOVE frying pan TO dish
(4) KEEP MOVE (keep last state of MOVE)

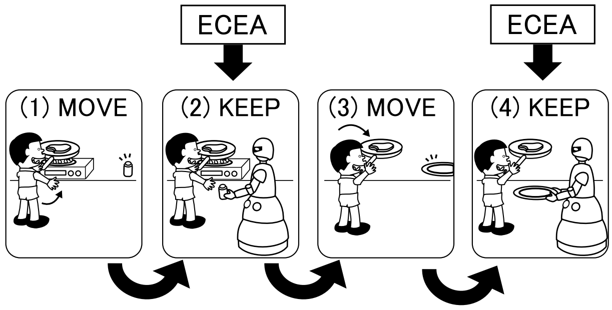

Fig. 19 ACS description with ECEA of a cooking task. ECEA is used in (2) and (4). When a user holds the hand during reaching the hand out for a spice bottle, a robot picks up the spice bottle and approaches to the user to pass it to her/him as shown in (2). When a user holds a frying pan during placing the food in a serving dish, the robot gives a dish to her/him as shown in (4).

In the experiment with participants, it required delicate instructions for participants. Therefore, it will need to adopt several evaluation methods to investigate the effect of the proposed method.

6 Related Work

Our approach is similar to studies in controlling robots by using gestures. Marrone et al. [9, 23] developed a domestic cleaning robot that is controlled by human gestures. For example, the robot cleans the area that the user designates. With this kind of approach, the user has to learn the gesture commands to precisely control the robot. In contrast, our approach does not require users to learn gesture commands for the robot. The robot using our method automatically cooperates with the users when they perform natural actions, such as moving obstacles.

In terms of natural actions, studies in controlling robots by force are similar to our concept. Hayashibara et al. [2] and Arai et al. [3] presented a robot that carried a long object cooperatively with a human. Nakai et al. [4] presented a control method for mobile robots that carried heavy objects. Yokoyama et al. [5, 6] studied a cooperative carrying task performed by a human and a humanoid. In those studies, the robots sensed the force of a user's push or pull on the object. It appears that force is interchangeable between robots and users through an object. In those interactions, commands to control the robots are embedded in the human actions. However, this method can be applied to limited tasks in which a user and a robot carry an object together. Our concept for a controlling method can be used for not only object carrying tasks but also other human-robot cooperative tasks, such as the sweeping task.

There have been studies on using human intentions for controlling robots. Terada et al. [24] developed an autonomous chair robot that behaves according to the human intentions based on computer vision technologies. Sato et al. [25] introduced an interface that assisted with human drawing tasks based on the user's intentional manipulation of the system. In general, intention inferring needs a lot of knowledge about the tasks and human models, but at a high processing cost. Although our approach appears to infer human intention, we are not attempting it. As our goal is to mediate human-robot cooperative tasks by using natural human actions, it does not always have to infer human intentions.

Our study is related to the studies in nonverbal communication between users and robots since our method uses commands embedded in a nonverbal human action. Kuniyoshi et al. [26] and Nicolescu [27] introduced robots that learn their behaviors by watching users' actions. The purpose of these studies is to make robots perform actions that humans previously performed. In our method, the robot does not learn new behaviors but is able to cooperate with the users.

Zhao et al. [28] proposed an implicit interaction between a user and robots. By their method, the user put cards to the object to be manipulated before the robot begins to perform. Their method is classified DCM because the user's action putting the cards to send commands to the robot is regarded as additional actions.

7 Conclusion

In this paper, we classified the conventional human-robot interaction into two groups: direct commanding method (DCM) and commands embedded in actions (CEA). We proposed Extended CEA (ECEA) based on CEA and its advantages are efficiency improved by not requiring additional human actions like CEA, and functionality to enable a robot to perform further actions like DCM. Users do not need to precisely manipulate the robot and to learn the manipulation method by using ECEA. Commands are sent to a robot according to a human action sequence that achieves their own task. In order to enable the robot to achieve tasks, we introduced a keep-based interaction in which a human keeps an action in the sequence for a certain period.

We applied ECEA to the cooperative sweeping task between a human and a robot. The robot was able to autonomously sweep the top of the desk using a random spinning strategy. When the robot found the lifted object by a user, it swept the area under the object. Users' physical workloads would not be increased because the proposed method does not require additional human actions. However, the effect of the proposed method on users' cognitive loads is unknown. An experiment to confirm the reduction of users' cognitive loads by using ECEA was conducted in a sweeping task. We measured human cognitive loads and compared ECEA with DCM. The experimental results showed that ECEA has a lower cognitive load than DCM. This suggests that ECEA was more suitable for achieving such a task than DCM.

In section 5.2, we showed three examples of ECEA applications. Based on the applications, we are currently planning to apply ECEA to other tasks in the domain of daily living situations. We believe that ECEA will assist users in noticing unknown robot function, because they only have to perform actions that occur unconsciously to them according to the given situation.

References

1. United Nations Economic Commission for Europe, International Federation of Robotics (2004) World Robotics 2004 – Statistics, Market Analysis, Forecasts, Case Studies and Profitability of Robot Investment. Palais des Nations
2. Hayashibara Y, Sonoda Y, Takubo T, Arai H, Tanie K (1999) Assist system for carrying a long object with a human – analysis of a human cooperative behavior in the vertical direction –. In: Proc. of 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems
3. Arai H, Takubo T, Hayashibara Y, Tanie K (2000) Human-robot cooperative manipulation using a virtual nonholonomic constraint. In: Proc. of 2000 IEEE International Conference on Robotics and Automation
4. Nakai K, Kosuge K, Hirata Y (2002) Control of robot in singular configurations for human-robot coordination. In: Proc. of IEEE Int. Workshop on Robot and Human Interactive Communication, pp 356–361
5. Yokoyama K, Maeda J, Isozumi T, Kaneko K (2001) Application of humanoid robots for cooperative tasks in the outdoors,. In: Proc. of IEEE/RSJ IROS Workshop on Explorations towards Humanoid Robot Applications

6. Yokoyama K, Handa H, Isozumi T, Fukase Y, Kaneko K, Kanehiro F, Kawai Y, Tomita F, Hirukawa H (2003) Cooperative works by a human and a humanoid robot. In: Proc. of the 2003 IEEE International Conference on Robotics and Automation, pp 2985–2991
7. Triesch J, von der Malsburg C (1997) Robotic gesture recognition. In: Proc. of the Bielefeld Gesture Workshop, pp 233–244
8. Waldherr S, Romero R, Thrun S (2000) A gesture based interface for human-robot interaction. Autonomous Robots 9(2):151–173
9. Marrone F, Strobel M (2001) Cleaningassistant - a service robot designed for cleaning tasks. In: Proc. of Advanced Mechatronic Systems
10. Breazeal C, Aryananda L (2002) Recognition of affective communicative intent in robot-directed speech. Autonomous Robots 12(1):83–104
11. Lauria S, Bugmann G, Kyriacou T, Klein E (2002) Mobile robot programming using natural language. Robotics and Autonomous Systems 38(3–4):171–181
12. Kajikawa S, Hiratsuka S, Ishihara T, Inooka H (2003) Robot position control via voice instruction including ambiguous expressions of degree. In: Proc. of IEEE Int. Workshop on Robot and Human Interactive Communication
13. Crandall JW, Goodrich MA (2002) Characterizing efficiency of human-robot interaction: A case study of shared-control teleoperation. In: Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems
14. Katagami D, Yamada S (2003) Active teaching for an interactive learning robot. In: Proc. IEEE Workshop Robot and Human Interactive Communication
15. Khamis AM, Rodríguez FJ, Salichs MA (2003) Remote interaction with mobile robots. Autonomous Robots 15(3)
16. Schwartz MF, Edwards ES, Montgomery M, Palmer C, Mayer NH (1991) The quantitative description of action disorganization after brain damage: A case study. Cognitive Neuropsychology 8:381–414
17. Newtson D (1973) Attribution and the unit of perception of ongoing behavior. Journal of Personality and Social Psychology 28(1):28–38
18. Newtson D, Engquist G (1976) The perceptual organization of ongoing behavior. Journal of Experimental Social Psychology 12:436–450
19. Newtson D, Engquist G, Bois J (1977) The objective basis of behavior units. Journal of Personality and Social Psychology 35(12):847–862
20. Choset H (2001) Coverage for robotics - a survey of recent results. Annals of Mathematics and Artificial Intelligence 31:113–126
21. Brooks RA (1986) A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation 2(1):14–23
22. Condron JE, Hill KD (2002) Reliability and validity of a dual-task force platform assessment of balance performance : Effect of age, balance impairment, and cognitive task. Journal of American Geriatrics Society 50:157–162
23. Strobel M, Illmann J, Kluge B, Marrone F (2002) Gesture recognition in a spatial context for commanding a domestic service robot. In: Proc. of the 33rd International Symposium on Robotics
24. Terada K, Nishida T (2002) A method for human-artifact communication based on active affordance. In: AAAI'02 Workshop on Intelligent Situation-Aware Media and Presentations
25. Sato T, Nishida Y, Ichikawa J, Hatamura Y, Mizoguchi H (1994) Active understanding of human intention by a robot through monitoring of human behavior. In: Proc. 1994 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 405–414
26. Kuniyoshi Y, Inaba M, Inoue H (1994) Learning by watching: extracting reusable task knowledge from visual observation of human performance. IEEE Transactions on Robotics and Automation 10(6):799–822
27. Nicolescu M, Mataric M (2001) Learning and interacting in human-robot domains. IEEE Transactions on Systems, Man and Cybernetics, Part A 31(5):419–430
28. Zhao S, Nakamura K, Ishii K, Igarashi T (2009) Magic Cards: A Paper Tag Interface for Implicit Robot Control. In: Proc. CHI '09, ACM Press, pp 173–182
29. Kemp C, Anderson C, Nguyen H, Trevor A, and Xu Z (2008) A Point-and-Click Interface for the Real World: Laser Designation of Objects for Mobile Manipulation. In: Proc. HRI '08, ACM Press, pp 241–248

Kazuki Kobayashi is an assistant professor at Shinshu University. He received his B.S. (2000) and M.S. (2002) in Media and Telecommunications Engineering from Ibaraki University and Ph.D. (2006) in Informatics from The Graduate University for Advanced Studies (Sokendai). His research interests include human-robot interaction, human-agent interaction, human-computer

interaction, artifact intelligence, and social intelligence. He is a member of IEEE, IEICE, JSAI, HIS, and RSJ.

Seiji Yamada is a professor at the National Institute of Informatics. Previously he worked at Tokyo Institute of Technology. He received B.S. (1984), M.S. (1986) and the Ph.D. (1989) degrees in artificial intelligence from Osaka University. His research interests are in the design of intelligent interaction including Human-Agent Interaction, intelligent Web interaction and interactive machine learning. He is a member of IEEE, AAAI, ACM, JSAI, IPSJ and HIS.