

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

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

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

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

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



Multimodal Command Language to Direct Home-use Robots

Tetsushi Oka
Nihon University
Japan

1. Introduction

In this chapter, I introduce a new concept, “multimodal command language to direct home-use robots,” an example language for Japanese speakers, some recent user studies on robots that can be commanded in the language, and possible future directions.

First, I briefly explain why such a language help users of home-use robots and what properties it should have, taking into account both usability and cost of home-use robots. Then, I introduce RUNA (Robot Users’ Natural Command Language), a multimodal command language to direct home-use robots carefully designed for nonexpert Japanese speakers, which allows them to speak to robots simultaneously using hand gestures, touching their body parts, or pressing remote control buttons. The language illustrated here comprises grammar rules and words for spoken commands based on the Japanese language and a set of non-verbal events including body touch actions, button press actions, and single-hand and double-hand gestures. In this command language, one can specify action types such as *walk*, *turn*, *switchon*, *push*, and *moveto*, in spoken words and action parameters such as *speed*, *direction*, *device*, and *goal* in spoken words or nonverbal messages. For instance, one can direct a humanoid robot to turn left quickly by waving the hand to the left quickly and saying just “Turn” shortly after the hand gesture. Next, I discuss how to evaluate such a multimodal language and robots commanded in the language, and show some results of recent studies to investigate how easy RUNA is for novice users to command robots in and how cost-effective home-use robots that understand the language are. My colleagues and I have developed real and simulated home-use robot platforms in order to conduct user studies, which include a grammar-based speech recogniser, non-verbal event detectors, a multimodal command interpreter and action generation systems for humanoids and mobile robots. Without much training, users of various ages who have no prior knowledge about the language were able to command robots in RUNA, and achieve tasks such as checking a remote room, operating intelligent home appliances, cleaning a region in a room, etc. Although there were some invalid commands and unsuccessful valid commands, most of the users were able to command robots consulting a leaflet without taking too much time. In spite of the fact that the early versions of RUNA need some modifications especially in the nonverbal parts, many of them appeared to prefer multimodal commands to speech only commands. Finally, I give an overview of possible future directions.

Source: Advances in Human-Robot Interaction, Book edited by: Vladimir A. Kulyukin,
ISBN 978-953-307-020-9, pp. 342, December 2009, INTECH, Croatia, downloaded from SCIYO.COM

2. Multimodal command language

Many scientists predict that home-use robots which serve us at home will be affordable in future. They will have a number of sensors and actuators and a wireless connection with intelligent home electric devices and the internet, and help us in various ways. Their duties will be classified into physical assistance, operation of home electric devices, information service using the network connection, entertainment, healing, teaching, and so on.

How can we communicate with them? A remote controller with many buttons and a graphical user interface with a screen and pointing device are practical choices, but are not suited for home-use robots which are given many kinds of tasks. Those interfaces require experiences and skills in using them, and even experienced users need time to send a single message pressing buttons or selecting nested menu items. Another choice which will come to one's mind is a speech interface. Researchers and companies have already developed many robots which have speech recognition and synthesis capabilities; they recognize spoken words of users and respond to them in spoken messages (Prasad et al., 2004). However, they do not understand every request in a natural language such as English for a number of reasons. Therefore, users of those robots must know what word sequences they understand and what they do not. In general, it is not easy for us to learn a set of a vast number of verbal messages a multi-purpose home-use robots would understand, even if it is a subset of a natural language. Another problem with spoken messages is that utterances in natural human communication are often ambiguous. It is computationally expensive for a computer to understand them (Jurafsky & Martin, 2000) because inferences based on different knowledge sources (Bos & Oka, 2007) and observations of the speaker and environment are required to grasp the meaning of natural language utterances. For example, think about a spoken command "Place this book on the table" which requires identification of a book and a table in the real world; there may be several books and two tables around the speaker. If the speaker is pointing one of the books and looking at one of the tables, these nonverbal messages may help a robot understand the command. Moreover, requests such as "Give the book back to me" with no information about *the book* are common in natural communications.

Now, think about a *language* for a specific purpose, commanding home-use robots. What properties should such a language have? First, it must be easy to give home-use robots commands without ambiguity in the language. Second, it should be easy for nonexperts to *learn* the language. Third, we should be able to give a single command in a short period of time. Next, the less misinterpretations, false alarms, and human errors the better. From a practical point of view, cost problems cannot be ignored; both computational cost for command understanding and hardware cost push up the prices of home-use robots.

One should not consider only sets of verbal messages but also *multimodal command languages* that combine verbal and nonverbal messages. Here, I define a multimodal command language as a set of verbal and nonverbal messages which convey information about commands. Spoken utterances, typed texts, mouse clicks, button press actions, touches, and gestures can constitute a command generally speaking. Therefore, messages sent using character/graphical user interfaces and speech interfaces can be thought of as elements of multimodal command languages.

Graphical user interfaces are computationally inexpensive and enable unambiguous commands using menus, sliders, buttons, text fields, etc. However, as I have already pointed out, they are not usable for all kinds of users and they do not allow us to choose among a large number of commands in a short period of time.

Since character user interfaces require key typing skills, spoken language interfaces are preferable for nonexperts although they are more expensive and there are risks of speech recognition errors. As I pointed out, verbal messages in human communication are often ambiguous due to multi-sense or obscure words, misleading word orders, unmentioned information, etc. Ambiguous verbal messages should be avoided because it is computationally expensive to find and choose among many possible interpretations. One may insist that home-use robots can ask clarification questions. However such questions increases time for a single command, and home-use robots which often ask clarification questions are annoying.

Keyword spotting is a well-known and popular method to *guess* the meaning of verbal messages. Semantic analysis based on the method has been employed in many voice activated robotic systems, because it is computationally inexpensive and because it works well for a small set of messages (Prasad et al., 2004). However, since those systems do not distinguish valid and invalid utterances, it is unclear what utterances are acceptable. In other words, those systems are not based on a well-defined command language. For this reason, it is difficult for users to learn to give many kinds of tasks or commands to such robots and for system developers to avoid misinterpretations.

Verbal messages that are not ambiguous tend to contain many words because one needs to put everything in words. Spoken messages including many words are not very natural and more likely to be misrecognised by speech recognisers. Nonverbal modes such as body movement, posture, body touch, button press, and paralanguage, can cover such weaknesses of a verbal command language. Thus, a well-defined multimodal command set combining verbal and nonverbal messages would help users of home-use robots.

Perzanowski et al. developed a multimodal human-robot interface that enables users to give commands combining spoken commands and pointing gestures (Perzanowski et al., 2001). In the system, spoken commands are analysed using a speech-to-text system and a natural language understanding system that parses text strings. The system can disambiguate grammatical spoken commands such as “Go over there” and “Go to the door over there,” by detecting a gesture. It can detect invalid text strings and inconsistencies between verbal and nonverbal messages. However, the details of the multimodal language, its grammar and valid gesture set, are not discussed. It is unclear how easy it is to learn to give grammatical spoken commands or valid multimodal commands in the language.

Iba et al. proposed an approach to programming a robot interactively through a multimodal interface (Iba et al., 2004). They built a vacuum-cleaning robot one can interactively control and program using symbolic hand gestures and spoken words. However, their semantic analysis method is similar to keyword spotting, and do not distinguish valid and invalid commands. There are more examples of robots that receives multimodal messages, but no well-defined multimodal languages in which humans can communicate with robots have been proposed or discussed.

Is it possible to design a multimodal language that has the desirable properties? In the next section, I illustrate a well-defined multimodal language I designed taking into account cost, usability, and learnability.

3. RUNA: a command language for Japanese speakers

3.1 Overview

The multimodal language, RUNA, comprises a set of grammar rules and a lexicon for spoken commands, and a set of nonverbal events detected using visual and tactile sensors

on the robot and buttons or keys on a pad at users’ hand. Commands in RUNA are given in time series of nonverbal events and utterances of the spoken language. The spoken command language defined by the grammar rule set and lexicon enables users to direct home-use robots with no ambiguity. The lexicon and grammar rules are tailored for Japanese speakers to give home-ues robots directions. Nonverbal events function as altanatives to spoken phrases and create multimodal commands. Thus, the language enables users to direct robots in fewer words using gestures, touching robots, pressing buttons, and so on.

3.2 Commands and actions

In RUNA, one can command a home-use robot to move forward, backward, left and right, turn left and right, look up, down, left and right, move to a goal position, switch on and off a home electric device, change the settings of a device, pick up and place an object, push and pull an object, and so on. In the latest version, there are two types of commands: action commands and repetition commands. An action command consists of an *action type* such as *walk*, *turn*, and *move*, and *action parameters* such as *speed*, *direction* and *angle*. Table 1 shows examples of action types and commands represented in character string lists. The 38 action types are categorized into 24 classes based on the way in which action parameters are specified naturally in the Japanese language (Table 2). In other words, actions of different classes are commanded using different modifiers. A repetition command requests the most recently executed action.

Action Type	Action Command	Meaning in English
<i>standup</i>	<i>standup_s</i>	Stand up slowly!
<i>moveforward</i>	<i>moveforward_f_1m</i>	Move forward quickly by 1m!
	<i>moveforward_m_long</i>	Move a lot forward!
<i>walk</i>	<i>walk_s_3steps</i>	Take 3 steps slowly!
	<i>walk_f_10m</i>	Walk fast to a point 10m ahead!
<i>look</i>	<i>look_f_l</i>	Look left quickly!
<i>turn</i>	<i>turn_m_r_30degrees</i>	Turn right by 30 degrees!
	<i>turn_f_l_much</i>	Turn a lot to the left quickly!
<i>turnto</i>	<i>turnto_s_back</i>	Turn back slowly!
<i>sidestep</i>	<i>sidestep_s_r_2steps</i>	Take 2 steps to the right!
<i>highfive</i>	<i>highfive_s_rh</i>	Give me a highfive with your right hand!
<i>kick</i>	<i>kick_f_l_rf</i>	Kick left with your foot!
<i>wavebp</i>	<i>wavebp_f_hips</i>	Wave your hips quickly!
<i>settemp</i>	<i>settemp_aircon_24</i>	Set the airconditioner at 24 degrees!
<i>lowertemp</i>	<i>lowertemp_room_2</i>	Lower the temperature of the room by 2 degrees!
<i>switchon</i>	<i>switchon_aircon</i>	Switch on the air conditioner!
<i>query</i>	<i>query_room</i>	Give me some information about the room!
<i>pickup</i>	<i>pickup_30cm_desk</i>	Pick up something 30cm in width on the desk!
<i>place</i>	<i>place_floor</i>	Place it on the floor!
<i>moveto</i>	<i>moveto_fridge</i>	Go to the fridge!
<i>clean</i>	<i>clean_50cm_powerful_2</i>	Vacuum-clean around you twice powerfully!
<i>shuttle</i>	<i>shuttle_1m_silent_10</i>	Shuttle silently 10 times within 1m in length!

Table 1. Action Types and Action Commands

Class	Action Types	Action Parameters
AC1	<i>standup, hug, crouch, liedown, squat</i>	<i>speed</i>
AC2	<i>moveforward, movebackward</i>	<i>speed, distance</i>
AC3	<i>walk</i>	<i>speed, distance</i>
AC4	<i>look, lookaround, turnto</i>	<i>speed, target</i>
AC5	<i>turn</i>	<i>speed, direction, angle</i>
AC6	<i>sidestep</i>	<i>speed, direction, distance</i>
AC7	<i>move</i>	<i>speed, direction, distance</i>
AC8	<i>highfive, handshake</i>	<i>speed, hand</i>
AC9	<i>punch</i>	<i>speed, hand, direction</i>
AC10	<i>kick</i>	<i>speed, foot, direction</i>
AC11	<i>turnbp, raisebp, lowerbp, wavebp</i>	<i>speed, body part, direction</i>
AC12	<i>dropbp</i>	<i>body part</i>
AC13	<i>raisetemp, lowertemp</i>	<i>room, temperature</i>
AC14	<i>settemp</i>	<i>room, temperature</i>
AC15	<i>switchon, switchoff</i>	<i>device</i>
AC16	<i>query</i>	<i>subject</i>
AC17	<i>pickup</i>	<i>width, height</i>
AC18	<i>place</i>	<i>height</i>
AC19	<i>push, pull</i>	<i>object, height, distance</i>
AC20	<i>slide</i>	<i>object, height, direction, distance</i>
AC21	<i>moveto</i>	<i>goal</i>
AC22	<i>switchcleaner</i>	<i>on-off</i>
AC23	<i>clean</i>	<i>area, repetition, mode</i>
AC24	<i>shuttle</i>	<i>distance, repetition, mode</i>

Table 2. Action Classes

3.3 Syntax of spoken commands

There are more than 300 generative rules for spoken commands in the latest version of RUNA (see Table 3 for some of them). These rules allow Japanese speakers to command robots in a natural way by speech alone, even though there are no recursive rules. A spoken action command in the language is an imperative utterance including a word or phrase which determines the action type and other words that contain information about action parameters. There must be a word or phrase for the action type of the spoken command, although one can leave out parameter values. Fig. 1 illustrates a parse tree for a spoken command of the action type *walk* which has *speed* and *distance* as parameters. The fourth rule in Table 3 generates an action command of AC3 in Table 2. The nonterminal symbol P3 corresponds to phrases about *speed* and *distance*. There are degrees of freedom in the order of phrases for parameters, and one can use symbolic, deictic, qualitative and quantitative expressions for them (see rules in Table 3).

There are more than 250 words (terminal symbols), each of which has its own pronunciation. They are categorized into about 100 parts of speech, identified by nonterminal symbols (Table 4). One can choose among synonymous words to specify an action type or a parameter value.

No.	Generative Rule	Description
1	$S \rightarrow ACTION$	action command
2	$S \rightarrow REPETITION$	repetition command
3	$ACTION \rightarrow AC3$	class 3 action command
4	$AC3 \rightarrow P3 AT3$	parameters and type (class 3)
5	$AT3 \rightarrow AT_WALK$	action type <i>walk</i>
6	$P3 \rightarrow SPEED$	phrase for speed
7	$P3 \rightarrow DIST$	phrase for distance
8	$P3 \rightarrow SPEED DIST$	speed + distance
9	$P3 \rightarrow DIST SPEED$	distance + speed
10	$DIST \rightarrow NUMBER LUNIT PE$	number + length unit + <i>PE</i>
11	$DIST \rightarrow DISTANCE_AMOUNT PE$	short, long
12	$P17 \rightarrow OBJECT17 HEIGHTKARA$	parameters for class 17 action
13	$HEIGHTKARA \rightarrow HEIGHTS KARA$	height of object to pick up
14	$HEIGHTS \rightarrow PLACE$	desk, floor, etc.
15	$HEIGHTS \rightarrow HEIGHT NUMBER LUNIT$	height in mm/cm/m
16	$HEIGHTS \rightarrow BODYPARTNO HEIGHT$	knee, hips
17	$OBJECT17 \rightarrow OBJWIDTH OBJECT$	object for class 17 action
18	$OBJWIDTH \rightarrow WIDTH NUMBER LUNIT NO$	width in mm/cm/m
19	$OBJWIDTH \rightarrow OBJSIZE$	small, large
20	$DIR \rightarrow DIR_DEICTIC PE$	deictic expression for direction
21	$REPETITION \rightarrow REPEAT$	repeat last action

Table 3. Example Generative Rules of RUNA

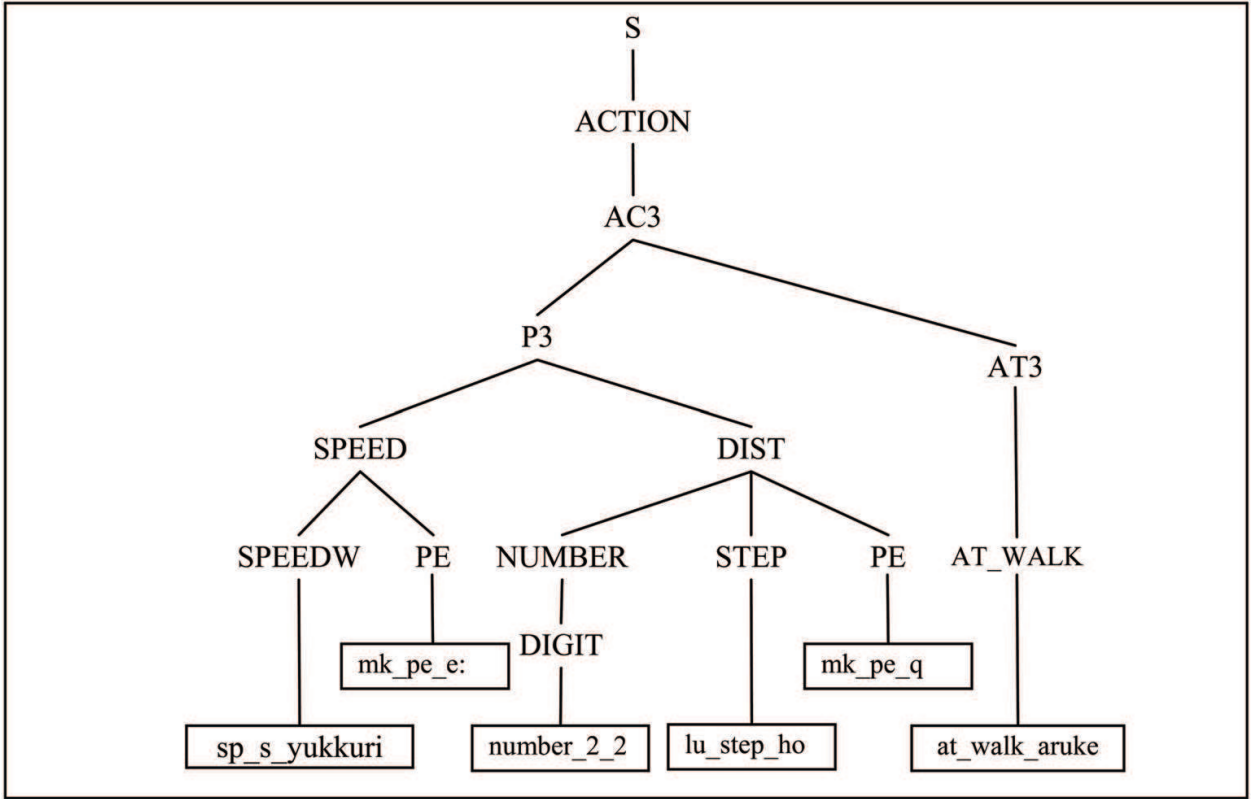


Fig. 1. Parse Tree for a Spoken Command (“Take, uh, two steps ... slowly!”)

Part of Speech	Words
AT_WALK	at_walk_aruke, at_walk_hoko, at_walk_hokosihro
GOAL	goal_refrigerator_reizoko, goal_entrance_iriguchi,...
ROOM	home_room_heyaa
KNEE	bp_knee_hiza
WIDTH	param_width_haba
HEIGHT	param_height_takasa
AROUND	param_around_shui
TIMES	param_times_kai
LUNIT	lu_mm_mm, lu_cm_cm, lu_m_m
TUNIT	tu_degree_do
DIR_LR	dir_r_migi, dir_r_migigawa, dir_r_miginoho, dir_l_hidari, ...
DIR_F	dir_f_mae, dir_f_zenpo
DIR_DEICTIC	dir_deictic_koko, dir_deictic_kocchi, dir_deictic_kochira
DIGIT	number_1_ichi, number_1_iq, number_2_ni, ...
SPEEDW	sp_f_hayaku, sp_f_isoide, sp_s_yukkuri, sp_m_futsuni
DISTANCE_AMOUNT	dst_long_okiku, dst_short_sukoshi, dst_short_chotto, ...
ANGLE_AMOUNT	ang_much_okiku, ang_little_sukoshi, ang_little_chisaku, ...
CLEANER_MODE	mode_powerful_zenryokude, mode_silent_shizukani
NO	joshi_no_no
NI	joshi_ni_ni
WO	joshi_wo_wo
PE (silence or hesitant voice)	mk_pe_q, mk_pe_a:, mk_pe_e:
REPEAT	md_repeat_moikkai, md_repeat_moichido

Table 4. Part of RUNA’s Lexicon

3.4 Nonverbal events

In RUNA, a set of *nonverbal events* is defined and used for commanding robots. These events are lists of character strings representing their own type and parameter values. Table 5 shows examples of nonverbal events. These events can be detected using sensors on home-use robots or buttons and sensing devices at users’ hand without much hardware and computational cost.

Event Type	Event Parameters	Example
<i>buttonpress</i>	<i>buttonid , iteration, duration</i>	<i>buttonpress_b4_3_124ms</i>
<i>bodytouch</i>	<i>position, iteration, duration</i>	<i>bodytouch_leftwrist_1_700ms</i>
<i>singlehandwaving</i>	<i>direction, iteration, stroke, frequency</i>	<i>singlehandwaving_left_3_long_120</i> <i>singlehandwaving_up_4_10cm_90</i>
<i>doublehandgesture</i>	<i>width, direction, iteration, stroke</i>	<i>doublehandgesture_wide_left_3_short</i>

Table 5. Nonverbal Events

3.5 Semantics

Since the language described above is syntactically unambiguous and simple, it is computationally inexpensive to identify action types and parameters in spoken commands.

As I have already mentioned, each spoken action command in RUNA includes a word specifying an action type, which can be distinguished by its own first string element *at* (Table 4). It can be divided into phrases expressing each parameter value and the action type using words which indicate the end of a parameter phrase, i. e. PE words (Fig. 1, Table 4). Therefore, it is straightforward and computationally inexpensive to identify the action type of a spoken command.

After a spoken command is divided into phrases and its action type is determined, a parameter value can be extracted from each phrase. It is always possible to determine which parameter the phrase is about by finding a keyword of a category such as LUNIT, AUNIT, DIR_LR, WIDTH, HEIGHT, and ANGLE_AMOUNT. If the keyword contains the parameter value, a string for the value of the parameter, *left* or *much*, is constructed. Otherwise, one must find a numerical expression to compose a string such as *1m* and *2degrees*. Thus, the spoken command in Fig. 1 is converted to a semantic representation *walk_s_2steps*.

Note that in RUNA there are deictic words and some parameter values can be left out in spoken commands. For instance, one may say “Turn slowly” without mentioning the direction or “Look this way” using a deictic expression perhaps with a gesture. In such cases, undecided parameters are resolved by nonverbal events described in the previous subsection. There are rules to map parameter values of nonverbal events (Table 5) to parameter values of action commands (Table 2). Designing these mapping rules is a key to a good multimodal command language that is natural and easy to learn. Table 6 shows examples of event parameters that correspond to action parameters.

If a spoken command has some parameters which cannot be resolved by nonverbal events, those parameter slots are filled with default values. Therefore, a command “Kick” is interpreted as “Kick slowly straight with your right foot” using the default parameter values for the action type *kick*.

Event Type	Event Parameter	Action Type	Action Parameter
<i>button press</i>	<i>button_id</i>	<i>turn / sidestep</i> <i>moveforward</i>	<i>speed, direction</i> <i>speed, distance</i>
	<i>iteration</i>	<i>raisetemp</i> <i>turn</i>	<i>temperature</i> <i>angle</i>
	<i>duration</i>	<i>turn</i> <i>walk</i>	<i>angle</i> <i>distance</i>
<i>body touch</i>	<i>position</i>	<i>kick</i> <i>raisebp</i> <i>turn / sidestep</i>	<i>foot</i> <i>bodypart</i> <i>direction</i>
<i>single hand waving</i>	<i>direction</i>	<i>turn / sidestep</i>	<i>direction</i>
	<i>stroke</i>	<i>walk</i> <i>turn</i>	<i>distance</i> <i>angle</i>
<i>double hand gesture</i>	<i>width</i>	<i>pickup</i>	<i>width</i>
	<i>iteration</i>	<i>pickup / place</i>	<i>height</i>

Table 6. Mapping between event parameters and action parameters

3.6 Command execution by home-use robots

A complete action command with its type and parameter values is executed by a home-use robot if the action is in the robot’s action repertoire. Quantitative parameter values in action

commands, e. g. *short*, are converted to quantitative values, e. g. *20cm* when robots execute the commands. The robot starts the action immediately if it has completed the previous command. Otherwise, the robot makes a decision depending on various conditions. It may start the new command immediately after completing the ongoing command, abort it and start the new command, or reject the new command explaining the reason. There is no good theory about the decision making yet, so we describe task specific rules for humanoids, robot cleaners, etc.

4. User studies

4.1 Objectives and methods

In the earlier part of this chapter I pointed out that a multimodal command language to direct home-use robots must have several properties. This opinion arises some fundamental questions:

1. Is the language easy for non-experts to learn and use?
2. How much time does it take to give a command in the language?
3. How expensive are robots that can execute commands in the language without significant delay and frequent misinterpretations?
4. How can the language be improved?

To answer these questions, one must collect data by conducting user studies that record multimodal commands by a wide range of users, speech recognition results, nonverbal events, system interpretations, reactions of home-use robots, user opinions, and so on.

The first question is about learning the command language. One can estimate a user's ability to give multimodal commands in the language (the user's linguistic performance) by giving various tasks. Fluency, human errors, command success rates, and time required for each command, and self-assessments can be indicators of performance. The second question is about the language's efficiency. One must investigate times required for commands by a wide range of users at several stages of learning. The third question can be answered by developing home-use robots and using them in user studies. The last question is related to the other questions and should be answered by finding all sorts of problems including human and system errors. Constructive criticisms by users also play a great role.

My colleagues and I have built a command interpretation system on a personal computer, small real humanoids (Oka et al., 2008), simulated humanoids, and a simulated robot cleaner that can be commanded in different versions of RUNA and conducted some user studies. In these studies, more than a hundred users, mostly young students, commanded one of the robots within 90 minutes. Some of them were asked to give spoken and/or multimodal commands printed in a sheet of paper. Many of them were given one or more goals to be achieved giving spoken or multimodal commands: checking a room, changing the settings of an air conditioner, moving a box, cleaning a dusty area, etc. We video-recorded the users and robots, recorded speech recognition results, nonverbal events, and command interpretations. Each user was asked to fill in a question sheet after commanding the robot.

Before asking each user to command one of the robots, we showed the person a short demonstration movie and handed a leaflet that illustrates how to command each action in diagrams and pictures (Fig. 2). We also prepared some short exercise programs to improve users' success rates and reduce human errors within 20 minutes.



Fig. 2. Parts of one of the leaflets which illustrate RUNA

4.2 Summary of results

In the user studies, the novice users were able to command our robots in RUNA consulting one of the leaflets and complete their tasks. In fact, there were many users who were able to direct the robot without their leaflet later in their tests.

Most of the users spoke clearly and fluently after practice, although there were a small number of hesitations, fluffs, and hashes especially in commands including more spoken phrases. Only several users made word misuses. In the latest studies, 92 - 98 % of spoken messages were correctly recognized with no word misrecognition thanks to the latest version (4.1.1) of an open source grammar based speech recognition engine (Lee et al., 2001). Most nonverbal messages were given properly after the users learned how to use them. There were few human errors such as pressing a wrong button, touching a wrong part of the robot body, and wrong gestures. However, there are problems in specifying some action parameter values in nonverbal messages. For example, most users made errors in choosing a length value out of five pressing a button, even after some practice to learn durations. There were also failures in specifying action parameters using hand gestures due to errors in our gesture detector using a web camera.

A majority of the users in the latest studies recorded a command success rate higher than 90 %. Most of user commands were completed within 10 seconds and our robots responded to them within a second or so. In fact, there were users who repeatedly gave multimodal commands very quickly without looking at the leaflet. Those users spoke immediately after pressing buttons or moving their hand(s) to the camera. About 77 % of the users who were asked a question about their preference answered that they preferred multimodal commands to speech only commands in RUNA. Those users selected multimodal commands to achieve their tasks more often than the others. In one of the latest studies, all of the 20 users felt that they understood how to direct robots, although some of them did not find it easy.

In the question sheets filled in by the users, there were some important opinions about the language. Some of them pointed out that it was difficult to learn to specify action parameter values in nonverbal messages. There were users who thought speech recognition errors caused problems for them.

4.3 Discussion

Further studies are needed to prove the effectiveness of the language for a wide range of non-experts, but our results imply that the current version of RUNA is fairly easy for Japanese speakers to learn. Although the novice users made some errors, they would not

need long time or much effort to fully master the spoken language and the set of nonverbal messages. With more experience, they would be able to give spoken commands specifying three or more action parameters fluently, use default parameter values whenever possible, and choose among nonverbal modes. As even novice users were able to command robots within a short period of time, experienced users would not take unnecessarily long time for a command.

Can users of home-use robots teach themselves the language? A demonstration movie and an introductory leaflet which illustrates examples of multimodal commands would help a novice user to grasp the principles of the language. Although it will take a while to master all types of actions, I suppose that it will be easier and easier to learn a new command.

Multimodal commands in the current version of RUNA can be interpreted using a microphone, a web camera, a controller or a keypad, tactile sensors, and a personal computer. Therefore, home-use robots in future would not need extra hardware for understanding commands in RUNA. Besides, more sophisticated speech recognisers and gesture detectors would reduce misinterpretations of user commands.

One should be able to make the language more natural and easier to learn by both amending the mapping between nonverbal events and action parameters and introducing new types of nonverbal events. In the current version of RUNA, some action parameter values are naturally mapped to event parameter values, and can be specified without acquiring skills. For example, it is easy for anyone to specify a direction by pressing a button, touching the robot, or using a gesture. Likewise, no special skill is necessary to give robots information about body parts, repetition counts, modes, and qualitative values such as *long* and *short*, in nonverbal messages. However, it is difficult for inexperienced users to specify angles, lengths, heights, and temperatures using buttons or gestures. There are three reasons for this. First, users need skills to specify quantities using durations, frequencies, or lengths; how long should I press the button to turn the robot by 30 degrees? Second, users must remember arbitrary mappings; how many times should I wave my hand to get a turn by 180 degrees? Third, our gesture detector cannot measure lengths with precision. This problem can be remedied by making use of a pen tablet, a touch panel, dials, or a screen to display parameter values. Another possible solution is to start an endless action and stop it by pressing a button, touching the robot, raising the right hand, saying "Stop," and so on. One should notice the existing methods are still helpful when users do not need high precision.

Word misuses found in the user studies prove the importance of choice of words for the lexicon of the spoken language. One can prevent word misuses by including as many Japanese words as possible. However, homonyms will increase risks of syntactically or semantically ambiguous utterances and speech recognition errors. Therefore, only frequent word misuses should be removed by adding new words to the lexicon.

5. Future directions

RUNA can be extended by adding grammar rules, spoken words, and types of nonverbal events. Without doubt, multimodal command languages to direct of multipurpose home-use robots in future must have more classes and types of actions. However, its framework based on type and parameter should work well for the purpose of giving home-use robots various kinds of action commands, goals, and missions despite the simplicity and limitations. Certainly, one must avoid syntactically or semantically ambiguous utterances

and select types of nonverbal events suitable for specifying parameter values of actions, goals, and missions taking into account both cost and usability.

Nonverbal messages can help human-robot communications in the same ways that they help human-human communications. They can not only segment and disambiguate verbal messages, but also convey the current status of humans and robots. Eye contacts, hand gestures, postures, body touches, and button press actions can be clues to detect and segment spoken commands, phrases, and words; paralanguage may play important roles in disambiguation; nonverbal messages that inform emotional and physical status may help robots' decision making.

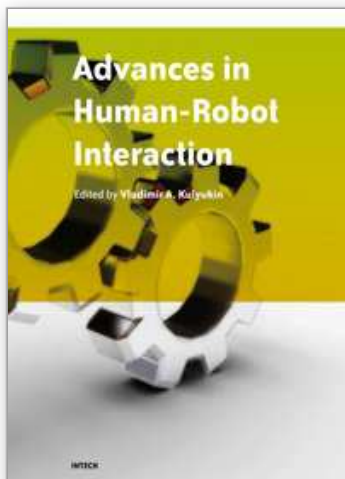
Multimodal languages for responses from home-use robots are also among my interests. Most importantly, robots can send nonverbal messages to convey their status and whether or not they can receive a new command at present. Another interesting future work would be fusing nonverbal and verbal messages. Redundant action parameter values in multiple modes may reduce risks of misinterpretations.

6. Acknowledgment

Development of RUNA was supported by KAKENHI Grant-in-Aid for Scientific Research (19500171). I would like to thank all my colleagues who worked on and discussed the subject with me at Fukuoka Institute of Technology.

7. References

- Bos, J. & Oka, T. (2007). Meaningful conversation with mobile robots. *Advanced Robotics*, 21., 1-2, (2007) 209-232, ISSN:0169-1864
- Iba, S.; Paredis, C. J. J.; Adams, W. & Khosla, P. K. (2004). Interactive multi-modal robot programming, *Proceedings of the 9th International Symposium on Experimental Robotics (ISER '04)*, pp. 503-512, ISBN:3-54-0288163, Singapore, March 2006, Springer, Berlin
- Jurafsky, D. & Martin, H. J. (2000). *Language and speech processing*, ISBN:0-13-122798, 2000, Prentice Hall, Upper Saddle River, New Jersey
- Lee, A.; Kawahara, T. & Shikano, K. (2001). Julius --- an open source real-time large vocabulary recognition engine, *Proceedings of the 7th European Conference on Speech Communication and Technology*, pp. 1691-1694, Aalborg, September 2001, International Speech Communication Association
- Oka, T.; Abe, T.; Shimoji, M.; Nakamura, T.; Sugita, K. & Yokota, M. (2008). Directing humanoids in a multimodal command language, *Proceedings of the 17th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN '08)*, pp. 580-585, ISBN: 978-1-4244-2213-5, Munich, August 2008, IEEE
- Perzanowski, D.; Schultz, A.C.; Adams, W.; Marsh, E. & Bugajska M. (2001). Building a multimodal human-robot interface. *IEEE Intelligent Systems*, 16., 1, (January-February 2001) 16-21, ISSN:1541-1672
- Prasad, R.; Saruwatari, H. & Shikano, K. (2004). Robots that can hear, understand and talk. *Advanced Robotics*, 18., 5, (2004) 533-564, ISSN:0169-1864



Advances in Human-Robot Interaction

Edited by Vladimir A. Kulyukin

ISBN 978-953-307-020-9

Hard cover, 342 pages

Publisher InTech

Published online 01, December, 2009

Published in print edition December, 2009

Rapid advances in the field of robotics have made it possible to use robots not just in industrial automation but also in entertainment, rehabilitation, and home service. Since robots will likely affect many aspects of human existence, fundamental questions of human-robot interaction must be formulated and, if at all possible, resolved. Some of these questions are addressed in this collection of papers by leading HRI researchers.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Tetsushi Oka (2009). Multimodal Command Language to Direct Home-Use Robots, *Advances in Human-Robot Interaction*, Vladimir A. Kulyukin (Ed.), ISBN: 978-953-307-020-9, InTech, Available from: <http://www.intechopen.com/books/advances-in-human-robot-interaction/multimodal-command-language-to-direct-home-use-robots>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2009 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen