Active Interface for Human-Robot Interaction

Nobuyuki Yamasaki and Yuichiro Anzai
Department of Computer Science, Keio University
3-14-1 Hiyoshi, Kouhoku-ku, Yokohama, 223 JAPAN
E-mail: {yamasaki, anzai}@aa.cs.keio.ac.jp

Abstract

In the near future, robots used by people like personal computers will appear in office or at home. In this paper, we call these robots personal robots. A personal robot can be thought as a small autonomous mobile robot. First, we discuss a user interface design for personal robots in the face-to-face situation. Second, we consider features of personal robots and their environments, and propose a new user interface concept for a personal robot: Active Interface. To show the effectiveness of Active Interface, we design and implement a speech dialogue system called Chaser for human-robot interaction based on Active Interface.

1 Introduction

Nowadays, high-performance computers connected by computer networks are coming into wide use. Many people use a computer to communicate with others through computer networks. And current research effort suggests that computer-supported cooperative work is important and will become popular.

But information exchanged during cooperative work is not only limited to electrical information which can be exchanged via computer networks. For example, physical objects such as documents or baggage are important information which should be exchanged during cooperative work. On the other hand, electrical appliances at home will become more intelligent. For example, a home cleaning robot is now made practicable [1].

In this way, it is very important for computing systems today in an office or at home to handle not only “the electrical world” with which is dealt by conventional computing systems but also “the physical world” in which we live.

So we use a personal robot to mediate from the electrical world to the physical world. In this paper, “personal” means personal use. We think a personal robot as a small general-purpose autonomous mobile robot, such as an office work support robot, a housework support robot, an amusement robot, a welfare robot and so on.

A personal robot can handle both the electrical world and the physical world. Using this ability, we try to expand the object handled by computer science to the physical world [2].

We believe that personal robots will eventually become members of our society in the near future, and co-exist with us in usual environments. A two-way communication between personal robots and human users is in needed: personal robots are expected to take orders from human users to accomplish assigned tasks, while in some occasions, personal robots may place explicit requests on human users for help. In this situation, we propose a new user interface called Active Interface to interact with human beings easily.

To show the effectiveness of Active Interface, we design and implement a speech dialogue system Chaser on the ASPIRE robot, which is a prototype of a personal robot, and we evaluate it.

2 Active Interface

2.1 Definition

We define Active Interface as follows. Active Interface does not only wait for users’ explicit input but also trials to get information from users’ implicit input and external environment. Based on the gathered information, it acts spontaneously and keeps the system in an advantageous condition for users (Fig.2).

We classify the input/output of Active Interface as follows.

- Explicit input: (key typing, mouse movement, voice, etc.)
- Implicit input: (facial expression[3], volume and direction of voice, etc.)
- Information from external environment: (location, noise level, temperature, etc.)
- Output: (action, display expression, utterance, etc.)

These are classified from an interface designer’s view. These are important for a designer in order to determine what kinds of input users should be aware of, and what kinds of input users can safely ignore.
In comparison between a conventional interface (Fig.1) and Active Interface (Fig.2), the former only uses explicit input. So user’s implicit information, which has certain amount of information, is lost. And the information from external environment has often been neglected in a conventional man-machine interface (especially human-computer interface). On the other hand, the latter actively handles and uses the information, which is discarded in conventional interfaces, and tries to keep the system in an advantageous condition for users.

![Figure 1: A conceptual model of a conventional interface](image)

![Figure 2: A conceptual model of Active Interface](image)

2.2 Design policy of Active Interface
Adapting to environmental changes
We must consider to adapt to environmental changes when we design a user interface of a personal robot, because the physical world changes dynamically in general. So we make a personal robot get environmental changes actively by various sensors, and we make the information reflected to its interface. We believe that it is not a user but a user interface that adapts to environmental changes. In this way, the user’s load is considerably lightened.

Applying personal robots’ own abilities
A personal robot has various abilities, such as moving itself, moving objects, etc. So these abilities may help to communicate with a user efficiently, when it interacts with the user directly without mediating a base station. For instance, using its moving ability, it can move to a place where it easily gets information from a user. And if it has an arm, it can touch a user physically by its arm and call upon the user’s attention, then exchange information with the user. In essence, the above approach is an extension of an active sensing[4].

Information from implicit input and external environment
Personal robots have various sensors to act in the physical world. Each sensor is also a device to get data from users, but it is difficult for users to understand and use it completely. So it is important how a designer of an interface lets users see sensors.

We make a point that users should not be aware of sensors as much as possible. First, we decrease what users should input to a personal robot explicitly. Second, instead of the above, we provide a new mechanism for personal robots to get information from users’ implicit input and external environment and use them actively. The mechanism can decrease the load of a user because the user does not need to input whole data explicitly.

We show an example of the above, which is a speech dialogue system. We assume that input from sensors is in various forms, such as voice, the volume, intonation, the direction of the sound source and the direction of the heat source.

First, we consider an interface design, when we simply introduce to a user that this robot obeys his orders (voice). When the user communicates with the robot, he may feel that he only gives attention to his voice as input. While the robot also gathers information from the volume and intonation as implicit input, and from the direction of the sound source and the heat source as information from external environment.

On the other hand, we consider an interface design, when we introduce to a user that this robot obeys his orders and moreover gets information from the volume of his voice, his intonation, the direction of his voice and his body temperature. This way increases his load in comparison with the first way, because he may think that he should pay attention to these various sensors.

Activities of Active Interface
The activities of Active Interface are not to urge a user to input something. If an interface has this type of activities, a user may feel it pushing. In this paper, we define the following two points as activities of Active Interface.

1. Active Interface does not only wait for explicit input from users but also gets information from implicit input and external environment.
2. *Active Interface* voluntarily acts to contact with
users advantageously as much as possible.

Many conventional interfaces only wait for input
from users. But using the first activity of *Active In-
terface*, a user does not need to input whole orders to
a system explicitly, so it may decrease the user’s load.
Moreover, using information from implicit input and
external environment together, it can get his explicit
input easily and precisely.

The second activity helps a user to input easily, and
makes a personal robot get information easily. Simi-
larly, this activity makes the personal robot tell him
information precisely.

3 Implementation of *Active Inter-
terface*

3.1 Applying to a speech dialogue sys-
tem

We consider the use of a speech dialogue system as
an interface of a personal robot. It is difficult for it
to recognize user’s orders (voice) with high precision,
because the environment around it will change dynam-
ically.

One of the solution is to let a user use a headset.
We think the method is not good, because it will make
him pay too much attention to I/O devices and feel
constrained. Instead, we use a directional microphone
as an input device and normal speakers as an output
device. This method can prevent the user from feeling
constrained. But it is weak in environmental changes
or noises.

3.2 Speech dialogue system: *Chaser*

Problems of the above system may malfunction by fol-
lowering environmental changes.

- Positional changes of a user (direction and dis-
tance)
- The volume changes of a speaker
- Noise level changes in the surroundings

A speech dialogue system needs a voice recognition
module and a voice synthesis module. The system
causes following problems by the above environmental
changes.

Voice recognition falling recognition ratio
Voice synthesis lack of information in noisy envi-
riment, giving a user an unpleasant feeling by
high volume in quiet environment

In recent years, some speech recognition systems
which are tolerant to noises are reported [5]. With

these techniques, applying *Active Interface* to a sys-
tem will prevent it from falling recognition ratio.

We try to apply *Active Interface* to a speech dia-
logue system named *Chaser* to solve these problems.
When the *ASPIRE* robot communicates with a user,
if noise level goes on rising, it will approach the user
until where it can get the high enough recognition ra-
tio. To design *Chaser* system, we classify I/O of the
system according to the I/O classification of *Active In-
terface*.

**Input**

- direct input from a user (explicit input)
- the volume of a speaker (implicit input)
- the direction of a user (information from external
  environment)
- the distance to a user (information from external
  environment)
- noise level in environment (information from ex-
  ternal environment)

**Output**

- moving (approaching or chasing a user)
- volume changes

3.3 The personal robot architecture: *ASPIRE*

We have already proposed a new personal robot archi-
tecture *ASPIRE* (*ASynchronous, Parallel, Interrupt-
based and REsponsive architecture*), which is designed
so as to move quickly in non-artificial and unknown
environment (ex. a crowd). And based on *ASPIRE*, we
have implemented the *ASPIRE* robot (Fig.3) as
a prototype of a personal robot [6]. The speech dia-
logue system *Chaser* was implemented on the *ASPIRE*
robot.

Figure 3: The *ASPIRE* robot
The **ASPIRE** robot is implemented as a functional classified parallel computer, which is connected by the VME bus and has distributed shared memories. It is divided into some functional modules. Each module has its own processing unit to process information and to decide its behavior for itself.

The most important characteristic of **ASPIRE** is that all I/O devices and all modules are systematically connected by interrupt lines (Fig.4). Using the interrupt lines, events can be transmitted immediately, only when necessary. (ex. The interrupt time between modules is 2.5 μsec.) So **ASPIRE** is good at both immediate and parallel processing. Because of the one-to-one correspondence between an interrupt and an event, **ASPIRE** can reactively handle both normal processing and exceptional processing as one kind of unified processing.

![Functional Modules](image)

Figure 4: Communication on the **ASPIRE** robot

To prevent the bottleneck caused by the shared bus, each module has two kinds of distributed shared memories (Fig.4). One is implemented by dual port memories (DPM), and the other is implemented by a bus controller and SRAM (Sh-Memory). The access time of DPM is faster than Sh-Memory, but the memory capacity of DPM is only 4 KB. On the other hand, the memory capacity of Sh-Memory is 1 MB. These distributed shared memories on its own module can be accessed without using the shared bus, so it is no problem to access the local distributed shared memory frequently. To communicate with other modules, each module writes data to the local distributed shared memory and interrupts the other module.

Each module always writes its own local status to its local distributed shared memory. The main module checks it periodically, and arbitrates among modules with conflicting behavior.

The following functional modules are implemented on the **ASPIRE** robot.

- **Main module** (MC68030, MC68882): Ethernet
- **Motor control module** (TMP68301): DC motor feedback control (PWM)
- **Sensor module** (TMP68301): US sensor, IR sensor, Touch sensor, Optical fiber gyroscope
- **Chaser module** (TMP68301, MC68882): Sound direction sensor, Heat direction sensor
- **Frame grabber module**
- **Image processing module** (T800 × 5)
- **Communication module** (included a main module)
- **I/O module** (TMP68301)
- **Voice synthesis module**
- **Voice recognition module**

The above modules have the operating system μ-PULSER [7] on board, which is a real-time, reactive, multitask and multithreaded OS. μ-PULSER integrates hardware interrupts and software interrupts using the mechanism called Direct Interrupt. This guarantees that the **ASPIRE** robot can act reactively.

### 3.4 Sensors for **Chaser**

In order to implement **Chaser** system, we must ensure that the **ASPIRE** robot has the ability to measure the direction as well as the distance of the user in advance, and the following sensors have been implemented.

**Sound source sensor**

In order to measure the direction of a user based on sound from the user, we have implemented a sound source sensor. This sensor measures the time difference of the sound that reaches to three microphones at the vertices of a right triangle with sides 30 centimeters long. The **Chaser** module computes the direction of the user based on the time difference similar to triangulation.

The **Chaser** module can get information like Fig.5 every one degree from the sound source sensor. Fig.5 shows the result of the experiment to evaluate the sound source sensor as follows.

1. Both a person and the **ASPIRE** robot stand still.
2. The person claps his hands at a regular time interval.
3. Based on these raw data, **Chaser** module (cf. section 3.5) computes the direction of the person similar to triangulation.
4. Each data in Fig.5 has been plotted by the computed data every one degree.
Heat source sensor
In order to measure the direction of a user based on heat from the user, we have implemented a heat source sensor made by a set of 8 direction pyroelectric infrared sensors on the ASPIRE robot (every 45 degrees). The pyroelectric element of the infrared sensor only reacts by moving heat source and does not react by stationary heat source, not to mistake stationary heat source (such as computers, lights, pots, etc.) for a human being.

The readings from the heat source sensor are synchronous to the readings from the sound source sensor. The Chaser module can get information like Fig.6 every 45 degrees from the heat source sensor. Fig.6 shows the result of the experiment to evaluate the heat source sensor. The data of the infrared heat sensor are digital, so the data in Fig.6 have been translated to the probability about time.

3.5 The Chaser module
We have implemented the Chaser module on the ASPIRE robot to control the above sensors. The Chaser module can reactively measure the direction of a user, using interrupt lines connected to all sensors according to ASPIRE and the computational ability of the co-processor (MC68882) on board.

Sensors used by Chaser system
Chaser system uses the following sensors.
- A speech recognition system (Panasonic VC-171)
- A speech synthesis system (NTT VCS-11)
- A noise level monitor [8]
- A sound source sensor
- 8 direction heat source sensor
- 8 direction US distance sensor
- 8 direction IR distance sensor
- 8 direction touch sensor
- An optical fiber gyroscope (Hitachi HOFG-3)
- 2 odometers (right and left moving wheels)

3.6 Approaching a user
The Chaser module can measure the direction of a user using the sound source sensor and the heat source sensor independently, but they may not be accurate owing to other sources such as reflected sound and other heat sources. The Chaser module can measure more accurately by considering the combined information. Fig.7 shows the combined information after simply fusing information of the sound source sensor (Fig.5) and the heat source sensor (Fig.6). Fig.7 shows that incorrect peaks are eliminated in comparison with Fig.5. As the result of the experiment, we decided that the Chaser module always uses the combined information every sampling. So the Chaser module computes the probability of the user’s existence from the combined information whenever sensor data is gotten, and determines the direction. Then it instructs the motor module to move to the direction.

The ASPIRE robot approaches a user as follows.
The sensor module measures the distance to a user using both the ultra sonic (US) distance sensor and the infrared (IR) distance sensor. An US distance sensor is used for the wide range (about 30 degrees) and the middle-to-long distance (about from 50cm to 5m). An IR distance sensor is used for the narrow range (about 5 degrees) and the short-to-middle distance (about from 5cm to 1m). The US distance sensor is used to find an obstacle or an object coarsely, and the IR distance sensor is used to measure the distance to the obstacle accurately.

The motor module navigates the ASPIRE robot to the direction computed by the Chaser module according to the distance measured by the sensor module.
Cooperating the other modules, the motor module approaches the user based on its odometers and information from them. In this way, if the motor module navigates the ASPIRE robot only based on the odometers, the errors caused by slips of tires or the like will be accumulated as time goes by. So the motor module corrects the global coordinates (an internal map) periodically (100 msec) using an optical fiber gyroscope controled by the sensor module. Because the precision of the optical fiber gyroscope is extremely high, it is no problem to use the ASPIRE robot in an office.

Even if the user moves to another place, the Chaser module can measure the direction of the user reactively and the sensor module can also measure the distance to the user based on ASPIRE. So ASPIRE robot can chase the user accordingly.

The above actions are arranged and maintained by the main module.

![Figure 7: The combined information](image)

4 Evaluation

In order to show the effect of making a system in an advantageous condition by applying Active Interface to the system, we performed two experiments using Chaser.

One is the experiment on measuring the direction of a user and approaching the user. The other is the experiment on the speech recognition ratio after the ASPIRE robot approaches the user.

4.1 Experiments

Approaching the user(s)

We investigated the success ratio of approaching a user using the sound source sensor and the heat source sensor independently or together.

We located sound sources and heat sources around the ASPIRE robot as following 4 cases. And we evaluated whether it can approach the user correctly.

Case 1: one sound source and one heat source
One person makes a sound.

Case 2: one sound source and two heat sources
One of the two persons makes a sound.

Case 3: two sound sources and one heat source
One person and one tape recorder make a sound simultaneously.

Case 4: two sound sources and two heat sources
Two persons make a sound simultaneously.

We regard the error angle range within 15 degrees toward the direction of movement as the correct approach in the experiments (each 30 times sampling).

Measuring the speech recognition ratio

We recorded white noise of FM radio broadcasting on tape and used it as a sample of noise in these experiments. We changed the noise level using the volume of a tape recorder and measured it by use of the noise level monitor based on JIS C-1502 A [8]. We experimented changing the noise level from −50dB to 0dB.

We used 11 words for the speech recognition. We have three persons pick up 10 words of them and produce a speech in sequence while we recorded the voice on tape. We used it on all experiments.

We changed both the noise level and the approaching distance.

4.2 Results and discussions

Approaching the user(s)

The result of the experiment is shown in Fig.8.

![Figure 8: Success Ratio of approaching the user(s)](image)
It was effective in cases of one heat source like case 1 or case 3 to measure the direction of a user only using the heat source sensor. However the success ratio was about 67% on average at most, because a heat source sensor had a wide range (45 degrees). In cases of two sound sources like case 2 or case 4, it fell down to 33% So we think that it is not effective only to use the heat source sensor.

It was effective in all cases except for case 4 to measure the direction of a user using information combining the sound source sensor and the heat source sensor. Using the method, the ASPIRE robot could measure the direction and approach the user with the success ratio of 95% on average in cases 1, 2, 3. Fig. 8 shows that the method is always superior to the others. Especially in case 3, the success ratio of only using the sound source sensor was 52% and the success ratio of only using the heat source sensor was 67%, while the success ratio of using the combined information was 96%. The results reveal that it is effective to use both sensors complementarily.

However in case 4 (two sound sources and two heat sources), no method could measure the direction exactly. We think that it is impossible to measure the direction of a user correctly using the current sensors.

Measuring the speech recognition ratio
The result of the experiment is shown in Fig. 9.

![Figure 9: Recognition ratio](image)

For example, to keep the recognition ratio of 50% (fat broken line) at the noise level of −30dB, the ASPIRE robot approaches the user until 23cm. If the noise level goes up to −20dB, the ASPIRE robot will actively approach the user until 17cm to keep the recognition ratio.

If a user want to keep the recognition ratio of 50%, the ASPIRE robot will try to approach or depart from the user according to the fat broken line in Fig. 9 by using Chaser.

The experimental results suggest that it is effective to measure the direction of a user and to approach the user, using Chaser, in the speech dialogue system in order to keep the recognition ratio.

5 Conclusion
In this paper, we proposed a new user interface concept Active Interface. And we designed and implemented the speech dialogue system Chaser on the ASPIRE robot based on Active Interface concept. The ASPIRE robot can measure the direction and distance of a user, and approach the user correctly by using various sensors. And using Chaser, the ASPIRE robot can approach and depart from a user when the noise level changes, and chase a user if the user moves. So Chaser can keep the recognition ratio at a substantial level. These experimental results of Chaser suggest that Active Interface concept is effective for human-robot interaction.

References