Human-machine Interface for Presentation Robot

Submitted 27th June 2011; accepted 27th September 2011

Jiri Krejsa, Vit Ondrousek

Abstract:
The purpose of mobile presentation robot is to interact with the users providing the required information. This paper presents the approach used in the highest software layer of autonomous mobile robot Advee used commercially for presentation purposes. The requirements for the layer are discussed together with available means of the robot. The particular building blocks and overall structure of the module are shown and key features are described in detail together with the behavior definition. Given module serves successfully when exploited to variable environment represented by people with computer literacy of great variation as proved during module verification tests.

Keywords: *human-robot interface, mobile robot, presentation robot*

1. Introduction

With mobile robots appearing nowadays outside research laboratories more than ever, human-robot interface (HRI) related issues attract great attention [1, 2]. Most of the research is focused on improving certain means of communication, especially the voice dialog (sound used both as inputs – natural language understanding and outputs – robot voice) [3, 4] and utilization of computer vision (human face recognition, face related higher features recognition, gesture recognition, etc.).

Various means of communication can be combined to increase the level of interaction. In [5] the speech recognition is combined with user localization using two processed microphones signals to obtain the location of the source, that can be further utilized in the response of the robot. Face recognition is accompanied with gesture recognition in [6] bringing wider variety of possible inputs to the robot. Work of Perzanowski [7] is an example of multimodal HRI, combining speech and gesture recognition with external input from Personal Digital Assistant (PDA).

This paper is focused on HRI of presentation robot Advee, whose purpose is to serve as an autonomous mobile source of information, transferable to people of different computer literacy. Such requirement brings the necessity of combining all robot means to get the redundancy in communication channels, so the less computer literate people can still get the message while more literate user is not repelled.

The paper is organized as follows. After short introduction of the robot itself the requirements for HRI layer are summarized together with available means (HRI related inputs and outputs). Proposed structure of the system is discussed with attention on key modules. The verification made during the tests together with discussion conclude the paper.

Fig. 1. Presentation robot Advee

2. Presentation robot

The purpose of presentation robot is to present certain information to the users in interactive manner. Further paragraphs describe the design of human-robot interface module for robot Advee. As the HRI is only a part of the robot system, here is a short introduction of the robot itself.

Presentation robot Advee is 160 cm high, 80 kg heavy wheeled robot with Ackerman steering. It has rear wheels driven with single DC actuator through mechanical differential. Robot is powered with 8cell LiFePo4 accumulator giving it approximately 8 hours operational range. The localization of the robot is handled by fusion of robot motion controllers commands, odometry readings and scanner of infrared beacons placed in known fixed locations. Fusion is performed by extended Kalman filter [8].

The software operated on the robot can be divided into three layers:

- low level provides interaction with hardware devices
- middle level implements robot state estimation and path planning
- high level handles the user interaction

The robot is equipped with a wide range of sensors and outputs, serving as the primary resources for HRI layer. The particular sources are described in detail below.

The size of the robot and access to its main communication devices is illustrated in Fig. 1.

3. HRI module

3.1. Module requirements

In order to design the HRI module properly the requirements must be stated first. The overall goal is to create the interface that is friendly and universal (can be used by variety of users), while the underlying engine is robust, modular and flexible. The requirements are listed in order of importance in Tab. 1.

Tab.1. HRI requirements

Requirement	**Description**
redundancy	information should be transferred to user in all possible ways in parallel (e.g. both visual and sound)
robustness	HRI module must be capable of operation even when minor hardware failure occur (e.g. camera fails)
flexibility	simple addition of new feature, sensor, etc
parametrization	overall behavior can be easily changed, certain features can be disabled/enabled upon request
adaptability	HRI should adapt to the user abilities, e.g. longer timeouts for slower users, etc.
modularity	several programmers can work independently on the project, particular modules can be tested independently

3.2. Robot means

Based on the requirements, what are the means of the robot? We can divide the means into two basic categories: inputs and outputs, where inputs represent the information from the user to the robot and outputs represent robot reaction. Some of the means are bidirectional, e.g. the touch screen acts both as input and output. The overview of Advee's means is shown in Tab. 2. and physical location of the means on the robot is shown in Fig. 2.

Tab. 2. Robot HRI means

Type / device	Details
Inputs	
Screen capacitive touch screen	robust touchscreen returning touch coordinates
Voice microphone, soundcard	incoming sound can be recorded and further processed
Vision CCD camera	instant flow of images further processed (face detection, etc)
Lower level bumpers, proximity sensors, odometry,	data from lower level of robot control (distances from obstacles, current location, etc.)
Outputs	
Screen LCD screen	hi resolution screen to present visual information
Voice soundcard, amp, speakers	modulated voice, currently prerecorded
Print printer	thermal printers up to 112mm wide
Motion motion actuators	safe motion in given area, towards given goal

Fig. 2. Robot means physical location

3.3. HRI building blocks

The key idea in designing the HRI module is to separate the interaction into independent blocks, that are sequentially activated by main HRI internal engine. Only single block is active at the time. The overall behavior of the robot can then be defined in a single routine, that controls blocks activation.

Each block uses different means of the robot, however in most cases the majority of the means are used.

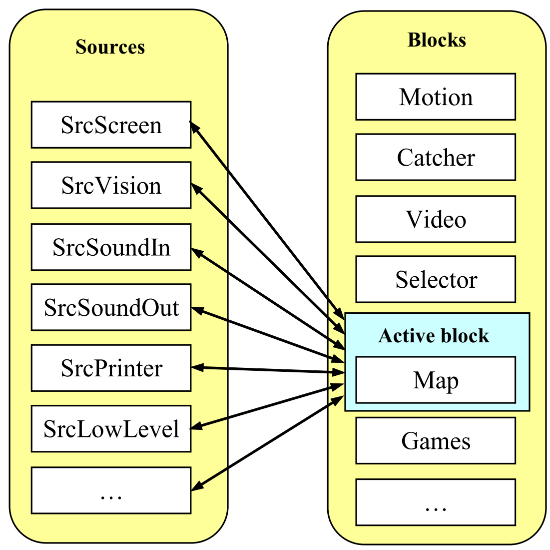

Fig. 3. HRI building blocks

To enable access to the means, the particular means are encapsulated into so called sources. Programmatically the blocks and sources are represented by certain classes. Each block is inherited from *BaseBlock* class and each source is inherited from *BaseSource* class. *BaseBlock* contains links to all the sources instances, therefore any subsequent block inherited can access all the sources. As the sources can serve to single block only, the mechanism that assigns and releases sources is implemented. Sources are assigned prior to block activation and released once the block is finished.

Currently implemented sources and blocks are shown in Fig. 3, with an example of *Map* as active block having access to all the sources. The sources directly depend on the means of the robot from previous chapter.

The sources encapsulate the access methods and events generated by corresponding devices. The key sources are:

Screen: shows the graphics and generates events when touch is detected.

Vision: processes the camera images with cascade of boosted classifiers based on Haar-like features for frontal face detection.

SoundIn: processes the incoming sound to detect simple patterns (no speech recognition yet)

SoundOut: plays prerecorded sound files of modulated voice. As the sounds are prerecorded, the variety of sounds is available for given type of the sound to avoid repeating the same phrases.

Printer: thermal printer with 112 mm wide output.

LowLevel: encapsulates the methods and events of lower level software layers. In particular the motion of the robot, its estimated position (given by Extended Kalman Filter based localization technique), the motion planner outputs, proximity sensors calls, hardware monitor calls (battery status, temperatures, etc.), etc.

Each block represents single operation in human-robot interaction. The key blocks are:

Motion – block that is active when the robot is moving and not interacting with people. Motion can end for a number of reasons: people are detected in surrounding (combination of proximity sensors analysis, face detection of the camera images, etc), somebody touches the screen, etc.

Catcher – block that verifies whether there is a person the robot can talk to. The block serves as decision maker for uncertain situations (low confidence in face detection, etc)

Selector – block that serves as the menu allowing user to select from several options, usually activating another block. The options are shown graphically as buttons on the screen with both pictograms and text. The selection is accompanied with voice explanation of the options.

Custom blocks – blocks responsible for certain feature of the data to present. The names are selfexplanatory, for example:

Map shows interactive map of the surroundings with indicated position of the robot in the map.

Video runs the video, interruption is allowed

Games enables users to play games, usually followed with the prints of reward in the case of victory in the game.

Print handles printing, guiding the user through the print process, checking whether user withdrawn the print, etc.

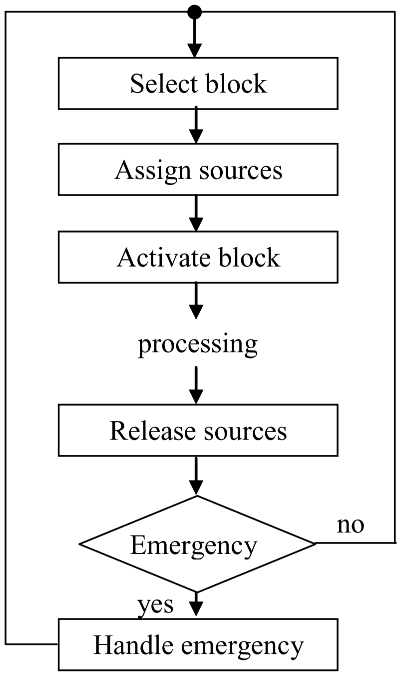

Fig. 4. HRI engine routine

Within each block a finite state machine is implemented, handling the behavior of the block, in particular the responses to user inputs (screen taps, incoming sounds, etc.) and internal events (end of the output sound, changes in lower level data, etc).

3.4. HRI behavior definition

Whole behavior of the robot is determined by the sequence of blocks. The sequence depends partially on user and partially on behavior definition. HRI engine works in simple cycle shown in Fig. 4. Once the appropriate block is selected, all sources are assigned to the block, block is activated and starts to operate. Within the block usually finite state machine is implemented, managing the incoming events and producing required outputs.

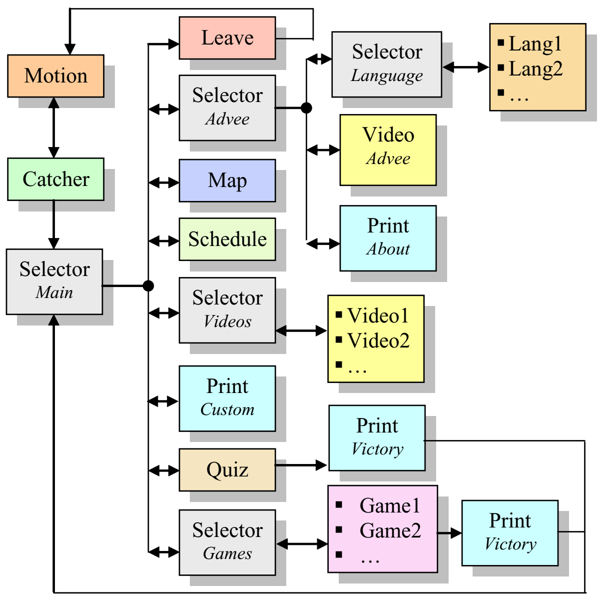

Fig. 5. Behavior signal flow

Once the block is done the sources are released and next block is selected.

Each block can finish in a regular manner, or once an emergency event occurs (HW failure, etc.). In such a case the emergencies are handled separately before the main cycle is entered again.

The behavior can be easily visualized with a flow chart, an example of the signal flow is shown in Fig. 5. The flow can be arbitrarily modified, certain features (blocks or block sequences) omitted or added.

3.5. Implementation details

High level layer of the robot software runs on Windows 7 OS. The HRI module is implemented in C# using Microsoft Visual Studio environment. Computationally demanding routines (e.g. image processing) are written in C and C++.

The predecessor to all sources classes – *BaseSource* class implements virtual methods for essential operations with the sources (startup and cleanup). The predecessor to all block classes – *BaseBlock* class implements virtual event responses for all possible sources events, therefore any descendant block can respond to such events. Virtual methods for block activation and deactivation are also implemented in *BaseBlock* class, taking care of assigning the sources to the currently active block and releasing the sources when block is finished. The termination of block processing is announced with *BlockDone* virtual event, overridden in particular blocks.

Lower layers of the software run on separate computer with Linux OS. Communication with the lower layers (position estimation, motion, path planning, etc.) is based on Lightweight Communication and Marshalling (LCM) library [9] and is fully encapsulated into *SrcLowLevel* source.

4. Verification

The HRI was initially verified using released version of HRI together with simulator of the lower layers of robot. During the tests the HRI was presented mainly to students and basic settings were tuned, such as timeouts for particular blocks, understandability of the graphics, etc. Once the basic setup was done the tests were performed using real robot on a number of people from various backgrounds and data in the form of questionnaires were collected afterwards together with the camera recordings.

Fig. 6. HRI in action

Based on the collected data the HRI was further modified, with the emphasis on redundancy of given information, e.g. all voice outputs (talk of the robot) are accompanied with the text on the screen of the same meaning. Further tests lead to minor modifications in rephrasing the speech output and changing the voice modulation for further clarification. During the use of the robot all the user inputs are logged and collected data are used for further analysis and improvements.

In order to determine whether the HRI module works according to requirements, several quantities can be extracted from data collected during the interactions. Such variables values obtained on several events with different types of users can be compared giving indication of HRI quality (ideally the quantities should not vary). While there could be numerous number of such indicators with respect to certain features (e.g. time spent with the map, win rate in games, etc.) only several give general impression on success of the HRI. Those indicators are described below.

ATI – average time of interaction with single user [sec]: interval between the start of the interaction and its end (user left the robot). Longer ATI means single user spent more time with the robot, it is only indirect indicator of how well the interaction went.

CSR – user catching success rate [-]: rate between successful catch of user (interaction starts) and failure (user walks away). Lower CSR means that people are afraid of the robot, do not know how to start an interaction, etc.

TIP – interaction timeout percentage [%]: percentage of interactions ended with timeout in arbitrary block other than main menu. Higher percentage means that user left the robot during interaction.

PPE – person per event [-]: total number of interactions divided by total number of users. If the PPE is higher than 1 then some users used the robot repeatedly.

BPU – blocks per user [-]: number of blocks activated per user in single interaction. Higher BPU means that user exploited more options during interaction.

Data collected on several events with different audience are listed in Tab. 3. The particular events were:

E1 – VIP audience
E2 – general audience
E3 – university students
E4 – seniors

Tab. 3. HRI indicators results

	e1	e2	e3	e4
ATI	48±11	73±19	85±21	96±71
CSR	0.89	0.75	0.97	0.18
TIP	21	13	14	8
PPE	1.12	1.12	1.51	1.05
BPU	5.2±1.2	7.3±2.1	8.7±2.4	3.2±1.9

Results clearly show that the type of the users does have an impact on given indicators values, however in none of the events the HRI failed. Very low user catching success rate in event 4 deserves further analysis. From the questionnaires given to the users we can state that the reason for the user to avoid the robot was not the HRI (users knew what to do), but general reluctance towards the robot itself.

5. Conclusion and future work

Described human-robot interface communication module was successfully designed and implemented in mobile robot Advee. HRI is now fully operational and used in commercial application of the robot with over 200 hours of operation in interaction mode with a variety of audience.

While HRI exhibits no problems, the parametrization and flexibility need improvements, that can probably be achieved only through the design of special script language for behavior description. Currently ongoing research is focused on extending the recognition capabilities (age, gender) of image processing unit. The gesture recognition was rejected as the cost of explaining how to use the gestures to the user overcomes the potential benefits.

ACKNOWLEDGEMENTS

Published results were acquired with the support of the CAS under the research plan AV0Z20760514 and with the support of the OPEC, project number CZ.1.07/2.3.00/09.0162. Authors would like to thank Petr Schreiber from Bender Robotics for his work on HRI coding.

AUTHORS

Jiri Krejsa* – Institute of Thermomechanics AS CR v.v.i., Brno branch, Technická 2, Brno, 616 69, Czech Republic, krejsa@fme.vutbr.cz.

Vit Ondrousek – Brno University of Technology, Faculty of Mechanical Engineering, Brno, 616 69, Czech Republic, ondrousek@fme.vutbr.cz

*Corresponding author

References

[1] M. W. Kadous, R. K.-M. Sheh, C. Sammut, "Effective user interface design for rescue robotics". In: *ACM Conference on Human-Robot Interaction*, Salt Lake City, UT, USA: ACM Press, 2006.
[2] K. Dautenhahn, M. Walters *et al*., "How may I serve you? A robot companion approaching a seated person in a helping context". In: *ACM Conference on Human-Robot Interaction (HRI)*, Salt Lake City, UT, USA: ACM Press, 2006
[3] M. Skubic, D. Perzanowski *et al.*, "Spatial language for human-robot dialogs", *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 34, issue 2, 2004, pp. 154-167.
[4] V. Kulyukin "Human-Robot Interaction Through Gesture-Free Spoken Dialogue", *Autonomous Robots*, vol. 16, no. 3, 2004, pp. 239-257.
[5] K. Park, S. Lee *et al*., "Human-robot interface using robust speech recognition and user localization based on noise separation device". In: *RO-MAN 2009*, pp. 328-333.
[6] M. Chang, J. Chou, "A friendly and intelligent human-robot interface system based on human face and hand gesture". In: *AIM 2009*, pp.1856-1861.
[7] D Perzanowski, A.C. Schultz *et al*., "Building a multimodal human-robot interface", *IEEE Intelligent Systems*, vol. 16, no. 1, 2001, pp. 16-21.
[8] J. Krejsa, S. Věchet, "Odometry-free mobile robot localization using bearing only beacons". In: *Proceedings of EPE-PEMC 2010 Conference*, Macedonia, pp. T5/40-T5/45.
[9] A.S. Huang, E. Olson *et al*., "LCM: Lightweight Communications and Marshalling". In: *IROS 2010*, Taipei, Taiwan, pp. 4057-4062.