Hibikino-Musashi@Home 2017 Team Description Paper

Sansei Hori, Yutaro Ishida, Yuta Kiyama, Yuichiro Tanaka, Yuki Kuroda, Masataka Hisano, Yuto Imamura, Tomotaka Himaki, Yuma Yoshimoto, Yoshiya Aratani, Kouhei Hashimoto, Gouki Iwamoto, Takashi Morie, and Hakaru Tamukoh

Graduate school of life science and systems engineering,
Kyushu Institute of Technology,
2-4 Hibikino, Wakamatsu-ku, Kitakyushu 808-0196, Japan,
http://www.brain.kyutech.ac.jp/~hma/wordpress/

Abstract. The team Hibikino-Musashi@Home was founded in 2010. It is based in Kitakyushu Science and Research Park, Japan. We have participated RoboCup@Home Japan open competition open-platform league every year since 2010. Currently, Hibikino-Musashi@Home has 24 members, who work at seven laboratories of Kyushu Institute of Technology. Our home-service robots are used as platforms for both education and implementation of our research outcomes. In this paper, we introduce our team and the technologies that we have implemented in our robots.

1 Introduction

Our team Hibikino-Musashi@Home, founded in 2010, competes in the RoboCup Japan open @Home open-platform league (OPL) every year. Our team is based in the Kitakyushu Science and Research Park and comprises 24 team members from seven laboratories of the Kyushu Institute of Technology. We are currently developing a home-service robot, and we use this event to present the outcomes of our research. In 2015 and 2016, we were placed third and second in the league, respectively. In addition, in 2016, we were awarded the first prize in Intelligent Home Robotics Challenge, which is a competition that takes place in Japan. This competition included a manipulation and object recognition test and a speech recognition and audio detection test that are same as the RoboCup@Home competition.

We have three objectives. The first is to participate in RoboCup@Home. This is a very important event for us because it gives us the chance to exhibit our research outcomes to other robot developers and research communities. The second objective is to develop a platform for research implementation. The members of Hibikino-Musashi@Home have a variety of research backgrounds because they come from seven different laboratories. Thus, Hibikino-Musashi@Home is able to merge the research results from these laboratories and test merged systems on robots. The third objective is to develop a platform for lectures. The Kitakyushu Science and Research Park has a joint graduate school intelligent car & robotics course[1] that is offered to both engineers and researchers human-resource development. Robots are used as one of the educational tools in this lecture program. Our team members can learn hardware- and software-development skills, collaboration skills that allow students to work with people outside of their own area of expertise, and team work.

This paper explains the hardware specifications and software systems as well as our scientific contributions to home-service robots.

2 Hardware

2.1 Overview of our robot platforms

We use two robots: Exi@ and Human Support Robot (HSR) [2]. Exi@ was developed by Hibikino-Musashi@Home in 2010. HSR was developed by the Toyota Motor Corp., and it has been in our team since 2016. Exi@ and HSR compete in the OPL and standard platform league (SPL), respectively.

Each robot has different characteristics that prove advantageous. Exi@, which bigger than HSR, can perform tasks at the same scale as humans. Furthermore, all the systems that Exi@ uses are developed by Hibikino-Musashi@Home. We have a thorough understanding of this robot, which makes it easy for us to implement research outcomes into it and apply new devices to it. Conversely, HSR is tiny and can be easily handled. Additionally, it is equipped with high-perfection hardware, and its behavior can be quite sophisticated.

In the following section, we introduce the basic hardware information for Exi@ as well as its development history.

2.2 Exi@

Figure 1 shows the appearance and history of Exi@ from 2011 to 2016. The first-generation Exi@ had one RGB-D camera, an arm, and a laser range finder (LRF) mounted on a robot base. In the 2012 model, we redesigned Exi@; this design was almost the same as that of the present design. The 2013 and 2014 models were given exterior shells to elicit better interactivity; these models were difficult to maintain and were very heavy. Thus, it became difficult to further improve Exi@. Consequently, the exterior had to be removed in the 2015 model. In the 2016 model, a vertical linear actuator camera was installed so that objects could be seen from the best position when detection. A field-programmable gate array (FPGA) was also installed in this model: it was added to handle intelligent processing requirements. Table 1 shows detailed information on the hardware of the current Exi@ robot.

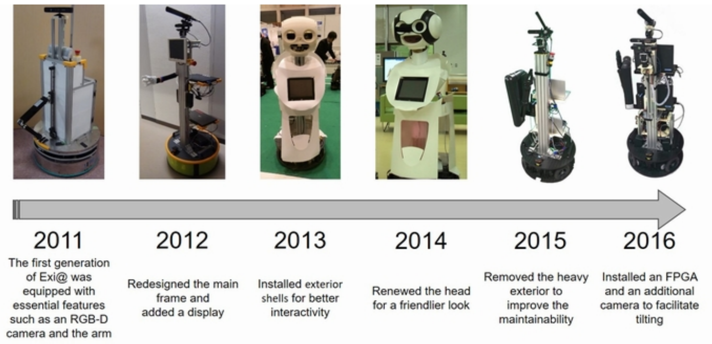

Fig. 1. Evolution of Exi@ from 2011 to 2016.

Table 1. Detailed information on the hardware used in Exi@.

Name	Exi@
Base	RoboPlus EXIA
Manipulators	Exact Dynamics iARM
LRF	Hokuyou UTM-30LX laser range finder
Microphone	SANKEN CS-3e Shotgun microphone
Batteries	Lead-acid battery 12V and 24V
Computer	ThinkPad PC Core-i5 4850U processor and 12GB RAM × 2
FPGA board	Xilinx ZedBoard[3] (FPGA + ARM processor)
Height	About 1500mm
Weight	About 80kg
Base size	About 600mm × 600mm

3 Software

3.1 Overview of the software

Figure 2 shows the software system used in Exi@: this system is based on Robot Operating System (ROS)[4]. ROS is one of the robot middleware, and it is the de facto standard middleware for the robot systems. Each software node used in Exi@ and HSR, e.g. image-processing and manipulator-control, uses a ROS interface to communicate with other softwares integrated in a robot system. We have developed a number of systems for Exi@, including image-processing, object-recognition, arm-control (for grabbing objects), simultaneous localization and mapping (SLAM), person-following, and sound-interaction systems, as well as a ROS-FPGA collaborative system. Information on the software that we use is provided in Table 2.

3.2 Object recognition

The process flow of the object-recognition system includes:

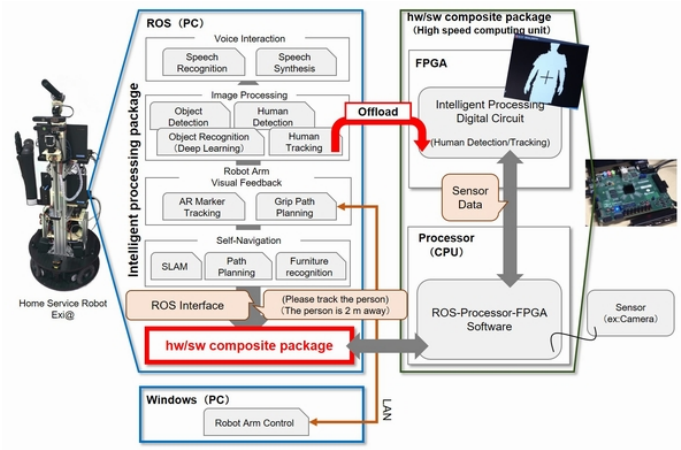

Fig. 2. Overview of the Exi@ software system.

1. Obtaining object images of objects using the RGB-D camera and the point cloud library (PCL)[5], which specialize in two-dimensional (2D) / three-dimensional (3D) point cloud processing.
2. Recognizing the object in the picture from the previous step using deep learning[6].

Hibikino-Musashi@Home uses a deep learning framework called Caffe[7]; this allows us to readily develop a deep neural network. The architecture of the neural network that our system uses is a 22-layered convolutional neural network called GoogLeNet[8]. However, the size of this network is too large for training all parameters used in a personal computer. Thus, we use transfer learning, which only trains the final layer of the network; in this training, we use images of the objects used in the RoboCup competition.

In RoboCup Japan Open @Home, a robot must be able to recognize 15 objects. Before the competition, we create a dataset that trains the deep neural network. We take 2,700 images of each object from various angles, and various amounts of noise are added to these images to improve our robots' object recognition accuracy.

3.3 Manipulator control

Exi@ is equipped with a manipulator produced by EXACT Dynamics known as iARM[9]. This manipulator is based on the premise that it is controlled by a human user operating a wheelchair. Therefore, the positioning accuracy of the iARM is not sufficiently good.

An augmented reality (AR) marker is attached to the arm, and an AR tracker is used in our system in order to improve control accuracy. The AR marker track-

Table 2. Detailed information on the software used in Exi@.

System	OS	Ubuntu 14.04
	Middleware	ROS Indigo
State management	SMACH (ROS)	
Voice interaction	Speech recognition (English)	Intel RealSense SDK 2016 R2
	Morphological Analysis Dependency Structure Analysis (English)	SyntaxNet
	Speech recognition (Japanese)	Julius
	Morphological Analysis (Japanese)	MeCab
	Dependency structure analysis (Japanese)	CaboCha
	Speech synthesis	Open JTalk
	Sound location	HARK
Image processing	Object detection	Point cloud library (PCL)
	Object recognition	Caffe with GoogLeNet
	Human detection / tracking	Depth image + particle filter
Robotic arm visual feedback	AR mark tracking	ar_track_alivar (ROS)
Self-navigation	SLAM	slam_gmapping (ROS)
	Path planning	move_base (ROS)

ing system is provided by ar_track_alvar[10], which is one of the ROS packages. Arm control is performed as the following:

1. Obtain the coordinates of the object using PCL and deep learning.
2. Move the arm into the vicinity of the object.
3. Use ar_track_alvar to detect the marker on the end effector using the RGB-D camera; feedback control is then performed to minimize the error with respect to the target coordinates.

3.4 Voice interaction

Voice interaction includes two systems that act as a voice recognition system and as a speech synthesis system. In addition, the Hibikino-Musashi@Home robots require both Japanese and English voice interaction systems and use the system that situation demands.

In the voice recognition phase, ambient noise blocks out a speaker's voice. To improve the voice-recognition accuracy, Exi@ uses a directional microphone and a microphone pan-tilt system that uses two servomotors; when listening to a speaker, the microphone turns toward it.

In a situation where a robot is being spoken to by someone whose location is unknown, the robot has to estimate the location of the speaker; our system uses HARK[11] and a microphone array to realize this.

3.5 ROS-FPGA system

Exi@ has a lot of intelligent systems implemented by software. Our robot has two lap-top computers to process the software demands required by these systems. However, these intelligent systems require a vast amount of computational resources and real-time processing to achieve a smooth interaction between Exi@ and humans. Thus, we installed an FPGA in Exi@. This allows some of the intelligent systems to be offloaded, which improves the overall processing speed. This is a big advantage for Hibikino-Musashi@Home when computing.

The advantages of FPGAs are that their internal digital circuit can be reconfigured and they process high parallelism, and consume a low amount of power. These advantages are ideal for the embedded systems used by home-service robots. However, the development period of FPGAs is significantly longer than that of software even if it is done by professional digital-circuit engineers. Developing an interface that can work with both ROS and FPGA is particularly difficult; therefore, using FPGAs can be problematic for robotics engineers that also wish to develop in software.

To overcome difficulty of this issue, we have proposed an novel interface called "connective object for middleware to accelerator (COMTA)". In our system, a hardware software (hw/sw) complex system comprising an embedded central processing unit (CPU) and an FPGA, provided by ZedBoard, is connected to a personal computer that execute ROS.

COMTA provides a common interface that enables easy access to FPGAs from ROS. Using this, we have been able to implement a person detection and following system into the FPGA that is connected to the main system by COMTA. In this experiment, we aim to improve its electrical efficiency and processing speed.

4 Scientific contributions

Currently, we are attempting to implement our research outcomes into our robots so that we can present them in the Final of RoboCup Japan open. In the Final, five highest-scoring teams in the competition demonstrate their systems. In this chapter, we introduce some of our technologies that have been demonstrated in the Final.

4.1 Operating a robot using brain wave

In RoboCup Japan open 2015, we proposed and demonstrated a method through which a robot could be commanded using brain waves. A user can command the

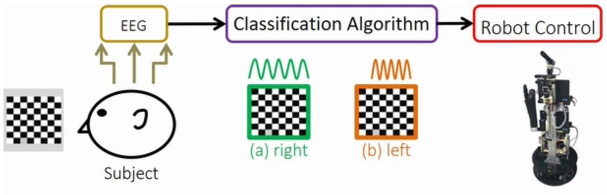

Fig. 3. System overview of the robot controller operated using brain waves.

robot using voice interactions or gestures, as in normally done; however, such actions are difficult for people that are unable to move, e.g., patients. Using brain waves to communicate with a robot would allow for a new interaction method.

Figure 3 shows an overview of the system we used for this. In our system, steady-state visual evoke potential (SSVEP) was used to control Exi@. The SSVEP wave frequency was changed by a subject observing the blink pattern displayed on the monitor, as shown in Fig.3 (a) and (b). We showed the subject two blink patterns that had different frequencies and then measured the SSVEP frequencies using an electroencephalograph. The robot was able to switch between the two actions which move to the right or left using the SSVEP frequency. In the future, this could be used as a system that patients can use to call for help.

4.2 Abnormality detection using a non-contact biosensor system

In RoboCup Japan open 2016, we demonstrated a system that could monitor a person's health; this system used a non-contact biosensor that was an outcome of research that had been conducted at our university. Figure 4 shows an overview of the system. In our demonstration, a robot measured the motion of a person's body and detected if they were falling. When the robot determines that a person is falling, it asks if the person is all right.

The non-contact biosensor that we installed irradiates the subject with radio waves and measures the reflection of the waves to measure vital information such as body motion, heart rate, and respiration rate. In addition, this sensor is able to measure this information through walls because it transmits radio waves. Therefore, the robot is able to monitor a subject located at a different place, e.g. if the robot is outside a bath-room that a user is currently occupying.

5 Conclusions

In this paper, we introduced the basic information and objectives Hibikino-Musashi@Home as well as the technologies used by the team. We have not only developed applications for the RoboCup@Home competition but have also developed applications for robots that are based on the research outcomes of our

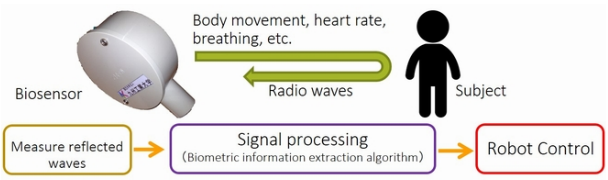

Fig. 4. Overview of the abnormality detection system.

institute. We intend to continue innovating deep learning technologies and ROS-FPGA interface, i.e., COMTA.

GitHub

Source codes of our systems are published in GitHub which URL is as follows: https://github.com/hibikino-musashi-athome

Acknowledgement

This work was supported by Ministry of Education, Culture, Sports, Science and Technology, the Joint Graduate School Intelligent Car & Robotics course (2012-2016), Kitakyushu Foundation for the Advancement of Industry Science and Technology (2013-2015), Kyushu Institute of Technology 100th anniversary commemoration projects, i.e., the student project (2015) and YASKAWA electric corporation project (2016), and JSPS KAKENHI grant number 26330279.

References

1. Joint graduate school intelligent car & robotics course. `http://jgs.kyutech.ac.jp/english/`.
2. Human support robot. `http://www.toyota-global.com/innovation/partner_robot/family_2.html`.
3. Zedboard. `http://zedboard.org/`.
4. Ros wiki. `http://wiki.ros.org/`.
5. Point cloud library. `http://pointclouds.org/`.
6. Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.
7. Caffe. `http://caffe.berkeleyvision.org/`.
8. Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.
9. iarm. `http://www.exactdynamics.nl/site/?page=iarm`.
10. ar_track_alvar. `http://wiki.ros.org/ar_track_alvar`.
11. Hark. `http://www.hark.jp/`.