Hibikino-Musashi@Home SPL 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. Our team, Hibikino-Musashi@Home, was founded in 2010. It is based in the Kitakyushu Science and Research Park, Japan. We have participated in the RoboCup Japan open @Home competition open platform league every year since 2010; however, this year will be our first time competing in the RoboCup @Home standard platform league. Currently, the Hibikino-Musashi@Home team has 24 members from seven different laboratories based in the Kyushu Institute of Technology. In this paper, we introduce the activities of our team and the technologies we have used for an intelligent home robotics challenge, which is a domestic Japanese competition.

1 Introduction

Our team, Hibikino-Musashi@Home, was founded in 2010, and it 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 we have 24 team members from seven different 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. This year, we will also participate in the standard platform league (SPL) of the RoboCup Japan Open@Home. In 2015 and 2016, we placed third and second, respectively, in the RoboCup Japan Open @Home OPL. Additionally, in 2016 we were awarded the first prize in the Intelligent Home Robotics (iHR) challenge, which is a competition that takes place in Japan. This competition included a manipulation and object recognition test as well as a speech-recognition and audio-detection test; these tests are the same as those generally used in the RoboCup@Home competitions. We used Human Support Robot (HSR)[1] in the iHR challenge 2016.

This paper describes the technologies that we used in the iHR challenge. In particular, this paper outlines both the image-recognition system that uses deep learning[2] and the voice recognition system installed in our HSR. For the iHR challenge, we used a customized open platform; however, customization is not

Fig. 1. Appearance of Human Support Robot (HSR).

permitted in the RoboCup@Home SPL. Nevertheless, we can contribute to the overall development of home service robot by applying the expertise we have developed with regards to such customizations.

2 Hardware

Hibikino-Musashi@Home has used an HSR as a standard-platform robot since 2016. We can customize this HSR as and when it is necessary. In this section, we introduce the specifications of our non-customized and customized HSRs.

2.1 Basic information

HSR we use, shown in Fig. 1, was developed by the Toyota Motor Corp. This robot is provided as an open-platform robot, and other universities and research institutions are also working on its development. Hibikino-Musashi@Home has used the expertise it has gained through the open-platform league as well as the software assets required for this platform in order to develop its HSR, and we are using this knowledge to develop highly functional robots. Table 1 outlines the detailed hardware information of our HSR.

We have successfully transplanted the “state management”, “voice interaction”, “image processing” algorithms related to “speech recognition and audio detection” and “object recognition and manipulation” which are RoboCup@Home competitions from a robot which is participated in OPL.

2.2 Customized model information

Hibikino-Musashi@Home uses a customized HSR, as shown in Fig. 2. We participated in the 2016 iHR challenge with it. The computational resources built into the HSR were not substantial enough to process our intelligent systems and exert the maximum level of the system performance. We therefore developed our own

Table 1. Detailed information about the hardware of our HSR.

Name	Human support robot (HSR)
Footprint	430mm
Height(min/max)	1005/1350mm (top of the head height)
Weight	About 37kg
Sensors on the moving base	Laser range sensor IMU
Arm length	600mm
Arm payload (recommended/max)	0.5/1.2kg
Sensors on the gripper	Gripping force sensor Wide-angle camera
Sensors on the head	RGB-D sensor x 1 Stereo camera x1 Wide-angle camera x1 Microphone array x1
Body expandability	USB x3 VGA x1 LAN x1 Serial x1 15V-0.5A output x1

Table 2. HSR external devices.

Computer	ThinkPad PC Core-i5 4850U processor and 12Gb RAM x2
Omnidirectional mic.	YAMAHA PJP-20UR
Egg-shaped mic. array	TAMAGO-03

customizations in order to solve this problem. As part of our customization, we added a laptop computer that could handle the requirements of our intelligent systems. We also added an egg-shaped microphone array that would facilitate sound localization, and we added an omnidirectional microphone to the HSR so that speech could be recognized. Table 2 shows the external devices attached to our robot. Consequently, the computer inside the HSR was able to be used solely for processing the HSR's basic software, such as its airframe control. This enabled the HSR to operate more stably. By adding in additional microphones, we were able to facilitate ambient sound recognition for our HSR.

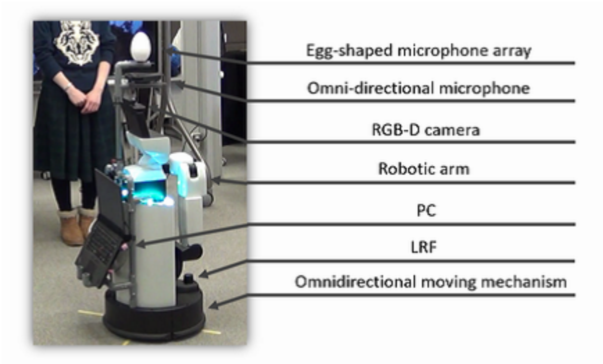

Fig. 2. Devices on the customized HSR.

3 Software overview

In this section, we introduce the systems that we installed in our HSR for the iHR challenge. We tackled the manipulation and object recognition test and the speech and audio recognition test by using an image recognition system that utilizes deep learning and a voice recognition system, respectively.

Figure 3 shows the customized system that we installed in our HSR. Our system is based on Robot Operating System (ROS)[3], which is the same infrastructure that we have used for our open platform league robot. As such, we have been able to rapidly develop a system for our HSR by porting over the software developed for our OPL robot, including our “state management”, “voice interaction”, and “image processing” algorithms.

In our customized HSR system, the laptop computer is used to process the system. This computer was connected to the computer embedded in the HSR through an Hsrb interface. The built-in computer specializes in low-layer systems, such as the HSR’s airframe control system and sensor system.

4 Manipulation and object recognition test

In this section, we explain the object recognition system employing deep learning that we used in the manipulation and object recognition test in the iHR challenge.

4.1 Processing flow of the object recognition system

Our object recognition system has two major parts, which are an object extraction system that obtains images from a camera, and an object recognition system. The entire processing flow of the object extraction and object recognition systems are as follows:

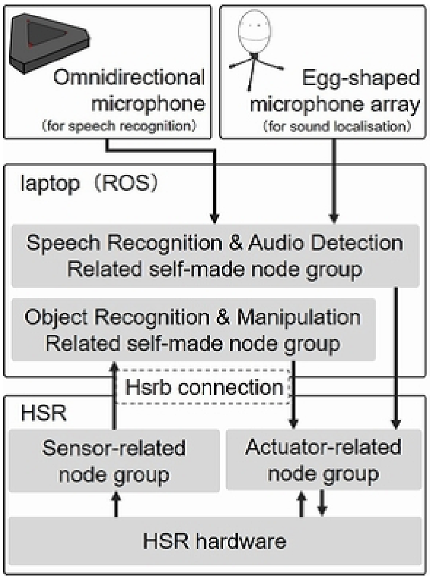

Fig. 3. Overview of the customized HSR software blocks.

1. Obtain a target 3D point cloud as shown in Fig. 4(b), by applying a pass through-filter to an original 3D point cloud as shown in Fig. 4(a) using a point cloud library (PCL). The image is obtained using a RGB-D camera on the HSR.
2. Apply a random sample consensus method, which includes the PCL, to the target 3D point cloud; this is used to detect the tabletop shown in the blue part of Fig. 4(c).
3. Clip an object 3D point cloud, which on the table, as shown in Fig. 4(d), from the target 3D point cloud.
4. Clip an object image from the RGB image using the object 3D point cloud in order to obtain an object image.
5. Classify the object using a deep neural network (DNN).
6. Calculate the object coordinates using the RGB-D camera coordinate system and the object 3D point cloud.
7. Change the object coordinates from the RGB-D camera coordinate system to the robot coordinate system.
8. Control a manipulator so that it moves toward the target coordinates and grasps the object.

4.2 Classification methods

We use Caffe[4], which is a deep learning framework and openCV[5], which is an image processing library, to construct the object recognition system. For

Table 3. Detailed software information used in HSR.

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
Self navigation	SLAM	slam_gmapping (ROS)
	Path planning	move_base (ROS)

the deep learning system, we use GoogLeNet[6], which is a 22-layered convolutional neural network that learns using the dataset provided by the Image Net Large Scale Visual Recognition Challenge 2012[7]. We employ transfer learning to GoogLeNet, and the final layer is fine-tuned by the dataset that we have created for RoboCup@Home. We used a training dataset that has images that are 224×224 pixels size, RGB-3Ch, and 8-bit gradation.

4.3 Experiments and results

Our object recognition system was trained using images of 15 objects, as shown in Fig. 5; these objects were used in the RoboCup Japan Open 2016, and they were used to evaluate the robots that competed. Images of these objects were captured from various angles while being rotated in order for a dataset to be created. After this, each pixel value of each image was multiplied by 0.9, 1.0, and 1.1 in order to add noise. Consequently, our dataset contained 2,700 images of each object. We used 2,000 of these images as the training data for the HSR.

Table 4 shows the results of our evaluation of the object recognition system with the test dataset, which included 12 objects which object number is one to twelve, as shown in Fig. 5. The test dataset was taken from 12 different directions at 30 degree increments in an environment that was different from the environment where the images for the training dataset were taken; this was done

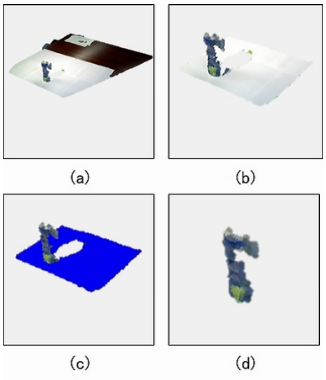

Fig. 4. Processing flow of the object recognition system.

Fig. 5. Objects used for the RoboCup Japan Open 2016.

Table 4. Classification results with GoogLeNet.

Method		Object number												Accuracy rate [%]
		1	2	3	4	5	6	7	8	9	10	11	12	
GoogLeNet2014	True	12	12	12	11	12	12	12	12	12	12	12	12	99
	False	0	0	0	1	0	0	0	0	0	0	0	0	

so that we could evaluate how the system responded to environmental changes. From the experimental results, we determined that our object recognition system performed to a high degree of accuracy for the RoboCup tasks.

5 Speech and audio recognition test

We customized the HSR by mounting omnidirectional microphone onto it along with an egg-shaped microphone array. The HSR voice recognition/sound source localization system operated as follows:

1. The omnidirectional microphone captures the voice of a person addressing it and uses a speech recognition engine, Julius[8], to recognize that it is being spoken to.
2. The voice of the person addressing it is captured by the microphone array, and a sound source localization is performed by the MUSIC method using HARK[9], which is a piece of auditory software for robots.

6 Results of the iHR challenge

We participated in the 2016 iHR challenge using our customized HSR. In the speech recognition and audio detection test, we scored 150 points out of 150. In the manipulation and object recognition test, we got 102.5 points out of 150. Thanks to these results, we were awarded first place in the competition.

7 Conclusions

In this paper, we introduced information on the HSR that participated in the 2016 iHR challenge; we also described its object recognition and voice interaction technologies. However, this customized HSR is unable to participate in the RoboCup@Home SPL, as customized robots are not allowed to compete in the league. Nevertheless, we can apply the expertise we gained from developing this HSR to non-customized HSRs, and we intend to use this knowledge to develop innovative technologies for home-service robot. Currently, we are developing a number of different pieces of software for non-customized HSRs, and we intend to participate in the RoboCup@Home SPL using a non-customized HSR.

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, 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 project : student project (2015) and YASKAWA electric corporation project (2016), and JSPS KAKENHI grant numbers 26330279.

References

1. Human support robot. `http://www.toyota-global.com/innovation/partner_robot/family_2.html`.
2. Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.
3. Ros wiki. `http://wiki.ros.org/`.
4. Caffe. `http://caffe.berkeleyvision.org/`.
5. Opencv. `http://opencv.org/`.
6. 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.
7. Imagenet large scale visual recognition challenge 2012. `http://image-net.org/challenges/LSVRC/2012/`.
8. julius. `http://julius.osdn.jp/`.
9. Hark. `http://www.hark.jp/`.