LiU@HomeWreckers 2019 Team Description Paper

Sofia Thunberg, Simon Wijk, Fredrik Löfgren, and Mattias Tiger.

Department of Computer Science and Information science
Linköping University, 581 83 Linköping, Sweden
http://www.ida.liu.se/divisions/hcs/coin/robocupathome/

Abstract. This is the application for the RoboCup@Home 2019 Social Standard Platform League from the team LiU@HomeWreckers. The main research line is in human robot interaction (HRI) with the goal for the Pepper robot to engage in experiments in real Swedish homes. This research is important for the social robotics community and triggers many other research areas, such as navigation and natural language processing. With the experience from last years competition, our team will focus on increasing the HRI skills and run more local software on the Pepper robot. The upcoming year's work will increase social acceptance of robots in the society. As a result the research team will able to work toward humanoid robots living in homes of elderly and people with cognitive disabilities.

1 Introduction

We are a RoboCup@Home SSPL team with strong ties to human computer systems research, and in particular human robot interaction (HRI) research. The competition is an opportunity for us to engage with the wider social robotics community to share and leverage knowledge and source code in a friendly atmosphere.

One of the central concerns in HRI research is how to make interaction with robots socially acceptable and comfortable to humans [4]. One of the challenges related to this issue is to find a good balance and consistency between robot behaviour and morphology. According to one classification, robot morphologies can be anthropomorphic (human-like), zoomorphic (animal-like), caricatured and/or functional [6]. Exploiting the tendencies of people to anthropomorphize – to ascribe human-like characteristics to things, such as animals or inanimate things – has been suggested as a means of facilitating interaction between humans and robots [5]. In contrast to this suggestion, it has also been shown that highly human-like robots can cause feelings of discomfort or "uncanniness" in people [11, 12]. Hence, robot morphology affects the acceptance of people of robots as interactive partners. Walters et. al. (2006) argues that robots that are aware of their social space increase the social acceptance by humans. They found that the participants preferred a robot approaching from the left or the right side and that they disliked a frontal approach [19]. Pepper (in an experimental comparison to the Aldebaran NAO robot) has been found to have a lower degree of human social acceptance toward the robot, with a significant lower result of being likeable, intelligent, safe and lifelike [18]. This result can be explained with that Pepper acts more unpredictable than NAO (with life mode on) and that Pepper is a larger and more human-like robot and can be perceived as more uncanny.

When the participants in a experiment situation think that they are talking to a computer, though they in fact are not, and the conversation is instead mediated by a human operator (the so-called wizard), it is called a Wizard of Oz-design (WoZ) [3]. There are several reasons for wanting to conduct WoZ-experiments. One is that computers are rigid and that people are flexible. In an experiment when the participant interact with a computer, the situation might change quickly in an unexpected way. If the experiment relies on a static system that cannot change, the situation will not seem natural to the participant, and therefore it is better with a wizard that can adjust.

This is a common employed technique in HRI, where the wizard are controlling the robot on a number of things, such as the robots movements, navigation, speech and gestures [16]. Thellman et. al. developed a WoZ system for Pepper using virtual reality to control the robot in a more natural way than with more traditional tools [17]. Researchers mainly uses WoZ in HRI because robots are not sufficiently advanced to interact autonomously with people in socially appropriate ways, and this method make it possible for the researcher to envision what future interaction could be like. Riek say that one methodological concern regarding WoZ is that one can argue that it is not really human-robot interaction so much as human-human interaction that is mediated through a robot. Another concern is the ethical problems that arise when the participant cannot tell whom they are interacting with, the robot or the test leader.

With all the stated issues regarding WoZ we are suggesting implementation with more autonomous features, such as:

- Face detection and recognition.
- Object detection and recognition.
- Gesture detection and recognition.
- Navigation, localisation and mapping.
- Speech recognition and generation.
- Manipulation.

These features would make it possible for a more natural robot interaction instead of a natural wizard interaction. This will have an impact on the research in natural environments, like in a home, a classroom or a retirement facility.

The long term goal with our participation will be to increase the social acceptance for Pepper in real world environments, to be able to do research toward elderly and people with cognitive disabilities.

During the coming years we will work on building a competitive software stack for Pepper which enable us to undertake interesting new research within the fields of HRI and artificial intelligence.

2 Team Background

The team consist of members from two research divisions under the Department of Computer and Information Science at Linköping University; the Human-Centered Systems (HCS) and the Artificial Intelligence and Integrated Computer Systems (AIICS).

HCS do research on distributed and situated cognition, cognitive ethnography, learning technologies, and design cognition. We also do research on artificial intelligence including knowledge representation, machine learning, and natural language processing.

For more information on research at HCS see: `http://www.ida.liu.se/divisions/hcs/index.en.shtml`

AIICS has a long history of research in artificial intelligence and its application to intelligent artefacts. AIICS has focused mainly on unmanned aerial vehicles but the techniques and technologies developed can also be applied to humanoid robots. Our research includes for example logic-based spatio-temporal reasoning over streaming data, automated task and motion planning, task allocation in multi agent systems, and localisation and navigation of robots.

For more information on research at AIICS see: `http://www.ida.liu.se/divisions/aiics/`

LiU@HomeWreckers competed last year in German Open in Magdeburg and in RoboCup@Home in Montreal. Some team members also have experience competing in the RoboCup Soccer Standard Platform League using NAO robots. The Linköping team *Linköping Humanoids*, sometimes called *LiU Humanoids*, has competed during the years 2014-2017. The @Home SSPL provide us with an opportunity to leverage experience from the Soccer Standard Platform League for the interesting and challenging tasks found in social robotics.

3 Re-usability and applications in the real world

We base our software stack on the Robot Operating System (ROS) [14] which is a well known middle ware in the robotics community. It is widely used by research groups and companies when doing research into AI-Robotics, allowing various groups to create standardised packages for various tasks and algorithms, and to share these packages with the wider community. We are using a stable version of ROS, ROS Kinetic. By using ROS and by following the ROS guidelines it is easier for us and others to re-use our components and software stack. We are dedicated to the idea of giving back to the community and we will consequently release our source code as open source after the competition if accepted.

The Pepper robot platform is used by Linköping University and its representatives to promote robotics and artificial intelligence in the society at large and in the corporate sphere. We make frequent use of Pepper (and Nao) in various exhibitions, expos, demonstrations, school-visits, seminars and so on. Software developed will be used both in settings where the robot will interact with the public as well as in future HRI research.

4 Research Focus

HCS is interested in the interaction aspect between humans and robots, and especially how the robots behaviour is perceived by the humans. We are further interested in learning how to design robots, in terms of appearance, body language, behaviour and dialogues, as to be as accommodating to humans as possible during human robot interactions in the future.

AIICS is interested in the integration aspect of the complex software required for autonomous humanoid robots. We are further interested in artificial intelligence research building upon a sophisticated robotic software stack capable of operating in unstructured human environments under the presence of humans in a safe way.

5 Approaches

We use several standardised components from the ROS community together with external services. In this section our approach in each subsystem will be presented.

5.1 Intelligence

The top-level AI is based on a Hierarchical State Machines (aka statecharts). We are using the ROS package SMACH to build the state machine in a standardised way. Most of the states are implemented as ROS actions. We have also done some experiments with Behavior Trees, and might use that internally for some actions.

5.2 Natural Language Processing

We are using Dialogflow from Google to both do the speech recognition and text analysis. We have trained different agents for the different tasks. All intermediate transcripts are displayed on Pepper's tablet, and the fulfilment text is visualised in a speech balloon at the tablet. The fulfilment text is also sent to the default speech synthesizer in naoqi. In this way we can easily update and change the chat-bot, we can also develop it without physical access to the robot.

Dialogflow will give us a context and intents for each recognised text. We are using these to control the robot's behaviour. For example if someone asks the robot to bring the apple from the kitchen, it will detect the intent "bring-somehing" and the parameters "apple" and "kitchen", and the state machine can activate correct states with the parameters passed along.

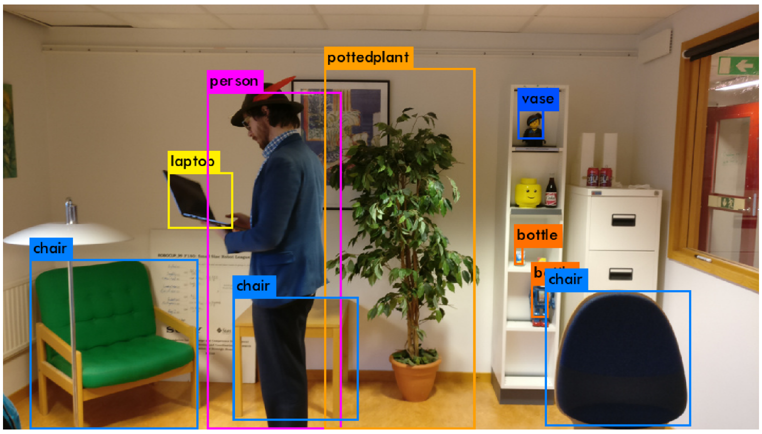

Fig. 1. Object detection and classification using the YOLO Detector with Darknet.

5.3 Vision

For face detection and recognition we use the Microsoft Face API.

Human and object detection is an important capability and we are utilising modern Deep Learning based approaches such as the YOLO Detector [15]. Figure 1 show an example of the YOLO detector output of our mock-up living room. We are further working on integrating state of the art pixel-based semantic segmentation [10].

Based on the detected objects from the YOLO module, tracking and record-keeping of objects are implemented. We are focusing on the tracking and remembering of persons, utilising a simple position-based reference matrix to estimate if two detected objects are the same. Our solution to the tracking of persons is implemented by combining data from the tracker with data from the Microsoft Face API, using the latter to remember human faces and thus determining if a detected person has been seen before or not, thereby facilitating HRI.

For the upcoming year we are interested in investigating if a probabilistic based model for object tracking, such as mentioned in [2] can be implemented successfully on the Pepper robot. We are also considering a series of comparison tests for face detection and recognition, analysing and comparing other solutions to Microsoft Face API.

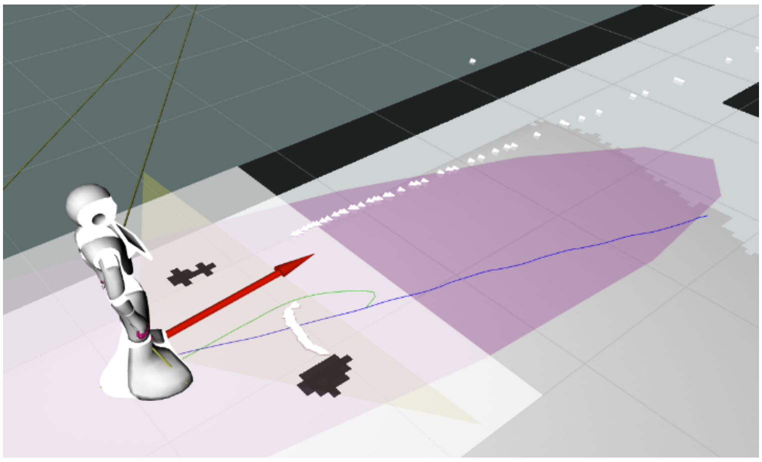

Fig. 2. Local planner (green line) and global planner (blue line) in a hallway, disturbed by two persons on each side in front of Pepper.

5.4 Navigation

A core problem to most tasks is exploration and navigation in a previously unseen environment. In order to facilitate such capabilities it is necessary to perform sufficiently good mapping of the environment and localisation within the environment using said mapping. We are evaluating several 2D and 3D mapping approaches, using both the laser sensors and the RGB-D sensor. We are investigating and comparing Google Cartagropher [8] and the ROS package gmapping when using both the RGB-D sensor and the laser sensors. We are looking into how to make these approaches work robustly for Pepper, where we for example compensate for the eye-lenses in front of the RGB-D sensor.

In order to navigate an environment, a map is needed. The map is created by using a mapping method as mentioned in the previous paragraph, or by measuring it by hand and drawing it in a suitable digital tool. Another key point is that both methods can be used simultaneously. With this in mind we may construct a map by drawing walls and stationary objects and at navigation-time take live data into consideration (such as a person or a dog). The process of navigating by using the aforementioned methods allows a robot to navigate in well-mapped areas as well as in a previously unknown territory. The map comes to play when the robot is given a navigation initiative. A global planner together with a local planner uses both static and dynamic data to plan a route from a point A to a point B. Whereas the global planner takes in already-known data to set a path, the local planner is subject to a changing environment. In reality, an attempt is made by the local planner to plan a path to join the global path while at the same time avoiding obstacles or other unforeseen events that affect the global path. To see the navigation process see Figure 2.

5.5 Knowledge

The knowledge base is an attempt to mimic the human memory. It gathers chosen information from the different topics of the Pepper robot, and saves it. When information is requested from the state machine, the knowledge base supplies it. The node database neo4j [13] saves the information, and a Python wrapper communicates with it. The advantage of a node based database is that connections can easily be created between each object, and that each object can contain a variety of different properties. Our main focus is to determine how the information sent to this module is interpreted and how it should be saved. Also how it should be extracted when the information is needed.

5.6 Human-Robot Interaction

The HRI skills are designed with a combination of lights in eyes and ears, sounds, touchscreen and talk. This is used both in developing and debugging processes but mainly for the competition scenarios. For example while developing navigation, we visualise the map on the touchscreen to see where the robot think it is. When debugging scenarios the robot says aloud where in the code it is so we easily can follow where it goes wrong. In the competition Pepper will ask the people to help the robot if necessary, for example in *Help-Me-Carry* Pepper follows a person and if it get lost, the robot will ask the person to stand in front of it again. In tasks like *EEGPSR* the robot will show in its eyes and ears different colours meaning that the robot is listening or processing the information. Also when giving the robot a task the robot will display the task on the touchscreen and ask to confirm that it has understood it correctly, either by touching its screen or by using voice.

6 Conclusions

The LiU@HomeWreckers team is interested in highly functional robots operating in unstructured human environments in the presence of humans. We are interested in how to integrate sophisticated components and build missing components to be competitive in the RoboCup@Home SSPL. We are further interested in using the resulting platform and software stack in order to do interesting human robot interaction research as well as artificial intelligence research. We have experience with integrating complex robotic systems, work with depth data for mapping and navigation, and human robot interaction. We consequently hope that we can contribute to the @Home SSPL and to be allowed to grow into a strong and competitive team.

References

1. Amos, B., Ludwiczuk, B., Satyanarayanan, M. (2016). "Openface: A General-Purpose Face Recognition Library with Mobile Applications," *CMU-CS-16-118, CMU School of Computer Science*, Tech. Rep.
2. Ardö, H,. (2009). "Multi-Target Tracking Using On-Line Viterbi Optimisation and Stochastic Modelling", *Lund University.*
3. Dahlbäck, N., Jönsson, A., & Ahrenberg, L. (1993). "Wizard of Oz Studies-Why and How". , 1–27.
4. Dautenhahn, K. (2007). "Socially Intelligent Robots: Dimensions of Human–Robot Interaction". *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362(1480), 679-704.
5. Duffy, B. R. (2003). "Anthropomorphism and the social robot". *Robotics and Autonomous Systems.*, 42(3), 177-190.
6. Fong, T., Nourbakhsh, I., & Dautenhahn, K. (2003). "A Survey of Socially Interactive Robots". *Robotics and Autonomous Systems*, 42(3), 143-166.
7. Honnibal, M., Johnson, M., (2015) "An Improved Non-Monotonic Transition System for Dependency Parsing", *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1373–1378
8. Hess, W., Kohler, D., Rapp, H., and Andor, D. (2016). "Real-Time Loop Closure in 2D LIDAR SLAM", *Robotics and Automation (ICRA), 2016 IEEE International Conference on. IEEE, 2016.* pp. 1271–1278.
9. Ioan A. Sucan and Sachin Chitta, "MoveIt!", [Online] Available: `http://moveit.ros.org`
10. Järemo Lawin, F., Danelljan, M., Tosteberg, P., Bhat, G., Khan, F. S., Felsberg, M. (2017). "Deep Projective 3D Semantic Segmentation", *17th International Conference on Computer Analysis of Images and Patterns (CAIP).*
11. MacDorman, K. F., & Ishiguro, H. (2006). "The Uncanny Advantage of Using Androids in Cognitive and Social Science Research". *Interaction Studies*, 7(3), 297- 337.
12. Mori, M. (1970). "Bukimi no tani [The Uncanny Valley]", *Energy*, 7 (4), 33-35.
13. [Online] Available: `https://www.neo4j.com`
14. Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., ... & Ng, A. Y. (2009, May). "ROS: an Open-Source Robot Operating System". *In ICRA Workshop on Open Source Software* Vol. 3, No. 3.2, p. 5
15. Redmon, J., Divvala, S. K., Girshick, R. B. (2016). "You Only Look Once: Unified, Real-Time Object Detection". *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* pp. 779-788.
16. Riek, L. (2012). "Wizard of Oz Studies in HRI: A Systematic Review and New Reporting Guidelines". *Journal of Human-Robot Interaction*, 1 (1), 119–136.
17. Thellman, S., Lundberg, J., Arvola, M., & Ziemke, T. (2017). "What Is It Like to Be a Bot? Toward More Immediate Wizard-of-Oz Control in Social Human-Robot Interaction". *In 5th International Conference on Human-Agent Interaction*, Bielefeld, Germany, October 17-20.
18. Thunberg, S., Thellman, S., & Ziemke, T. (2017). "Don't Judge a Book by its Cover: A Study of the Social Acceptance of NAO vs. Pepper. 5th International Conference on Human-Agent Interaction". *5th International Conference on Human-Agent Interaction*, Bielefeld, Germany, October 17-20.
19. Walters, M., Dautenhahn, K., Woods, S., Koay, K., Te Boekhorst, R., & Lee, D. (2006). "Exploratory Studies on Social Spaces Between Humans and a Mechanical-Looking Robot". *Connection Science*, 18 (4), 429–439.

Robot's External Services

For our Pepper robot we are using the following external services:

– **Google DialogFlow:** Python Dialogflow API, `www.googleapis.com/auth/cloud-platform` as endpoint together with a service account .json file.
– **Microsoft Face API:** Used for face recognition and other characteristics e.g. gender. Accessed via the URL `https://northeurope.api.cognitive.microsoft.com/face/v1.0/` and a subscription key.

Robot's Software Description

For our Pepper robot we are using the following software:

– **Navigation, localisation and mapping:** Cartographer from Google or Octomap + gmapping.
– **Face recognition:** Microsoft Face API.
– **Speech recognition, processing and generating:** Google Cloud Services.
– **Natural Language Processing:** Dialogflow from Google.
– **Object recognition:** YOLO Detector with Darknet.
– **Arms control and two-hand coordination:** MoveIt.
– **Gesture recognition:** OpenNI.
– **Knowledge representation:** neo4j (py2neo).
– **Decision making:** SMACH.
– **General programs:** ROS Kinetic, Choregraphe 2.5.10, Python 2.7, NAOqi 2.5.10 and OpenCV 3.3.0.