Abstract. This paper presents the b-it-bots@Home team and its mobile service robot called Jenny – a service robot based on the Care-O-bot 3 platform manufactured by the Fraunhofer Institute for Manufacturing Engineering and Automation. In this paper, an overview of the robot control architecture and its capabilities is presented. The capabilities refer to the added functionalities from research and projects carried out within the Bonn-Rhein-Sieg University of Applied Science.

1 Introduction

The b-it-bots@Home team was established in 2007 and is mainly driven by the international master’s program in Autonomous Systems offered at Hochschule Bonn-Rhein-Sieg (HBRS)\(^1\). Our team consists of bachelor’s, master’s, and PhD students, all of whom are advised by three tenured professors, such that we have a long history of participation at RoboCup@Home competitions.

The team’s first robot for the @Home league was a differential drive robot based on the VolksBot platform developed by Fraunhofer IAIS. From 2008 to 2010, this platform was used to successfully participate at the RoboCup German Open (2nd place in 2008, 1st place in 2009 and 2010) and RoboCup (2nd place in 2008, 1st place in 2009, and 3rd place in 2010). In 2011, the team acquired a Care-O-bot 3, which was developed by Fraunhofer IPA, and successfully migrated all existing software onto the new ROS-based platform\(^2\). Since 2011, the b-it-bots@Home team has successfully participated in multiple competitions:

\(^1\) http://www.h-brs.de
\(^2\) http://www.ros.org
RoboCup (3rd place in 2011, 9th place in 2016), RoboCup German Open (3rd place in 2011; 2nd place in 2012), and RoCKIn (2nd place - speech understanding functionality benchmark - in 2014).

While participation at competitions has always been an integral part of our activities, we foster a research-oriented culture above all, such that we have several ongoing PhD and master’s theses projects that are very closely related to the team.

The remaining sections present some of the work that current and previous team members have developed, focused on some of our research interests: execution monitoring, fault detection and diagnosis, robust manipulation, real-time and adaptive perception, as well as knowledge-based reasoning.

2 Task Planning

In order to enable robots to robustly handle unexpected situations, Awaad et al. [1] [2] have introduced three reasoning phases that use functional affordances to enable robustness and flexibility in robot task planning: the first phase generates a focused planning problem; the second phase expands the domain where necessary; the third and final reasoning phase uses affordances during plan execution and monitoring.

A task planning, execution, and monitoring system was developed by Shpieva and Awaad [3]. The planner used here was implemented using JSHOP2, which is a hierarchical task network planner; which is currently being integrated on Jenny.

In addition, the team also researches strategies to deal with external faults, such as execution failures caused by factors that a robot cannot control directly. Examples of such are (i) a logic-based qualitative reasoning model that uses naive physics for discovering the likely causes of external faults [4] and (ii) a simulation-based approach that can be used both for describing the desired behaviour during action execution and for improving execution models in case of remaining failures [5].

3 Robust Manipulation

3.1 Grasp Domain Definition Language

A Grasp Domain Definition Language (GDDL) [6] has been defined in this context, which can be used to specify complex task-oriented grasps. GDDL models allow a constraint-based optimisation problem to be derived automatically, such that the constraints enable validation of the specification and provide feedback during the design phase and run time.

3.2 Slip detector

Sanchez et al. [7] developed a slip detector that combines inputs from tactile sensors on the Care-O-bot’s hand and force/torque sensors on the arm of the
3.3 Reactive Placement

Safe object placement on a flat surface is achieved by detecting a contact between the object and the environment (e.g., a table). The placement procedure starts by moving the manipulator to a position directly above the desired placement location. This action is followed by the manipulator descending while monitoring the force-torque sensor. The output of the force-torque sensor is smoothened using a moving-average filter, as implemented in [8], and the filtered output is compared to a force threshold. The manipulator’s motion is stopped if the monitored force exceeds the threshold value and the hand’s fingers are opened to release the object. The advantages of using this approach allow the robot to overcome limitations in the accuracy of visual systems and to handle changes that might occur in a dynamic environment. More details about this approach can be found in [9].

![Fig. 1: Reactive placement](image)

3.4 Haptic Interface

The manipulator of the robot platform can be utilized to function also as a haptic interface for human users [8]. This functionality has been developed and implemented in two scenarios, guidance and cooperative transportation. In guidance, a user can control the robot’s base movement through interacting with the manipulator wrist or gripper. In the second scenario, cooperative transportation, the robot can carry an object together with a human user and follow the movement direction. Both scenarios are shown in Figure 2. Through the use of smoothing filter and PID controller, the feature is able to accommodate noisy input and produce a steady movement. The feature is developed so that it can be applied in
almost all possible configuration. Specifically for guidance scenario, a user trial have been performed and the results shows that the functionality is intuitive and compatible for different type of users.

![Fig. 2: Manipulator as a haptic interface](image)

4 Motion Detection

Mobile robots need to be able to perform perception tasks even when they are moving. One such task is motion detection, which is useful for obstacle avoidance, saliency detection, and so forth. A motion detection method that compensates for egomotion was developed by Thoduka et al. [10] based on the Fourier-Mellin transform (FMT) for compensating camera motion and temporal differencing for motion detection. The algorithm is able to run close to real-time on a robot and was integrated on our Care-O-bot robot.

![Fig. 3: Camera motion and independent motion (yellow circle) between frames](image)
5 Emotion and Gender Recognition

A real-time vision system is used to perform the tasks of face detection, gender classification and emotion classification simultaneously in a single step. The facial expression recognition system present in our robot, can recognize human expressions of joy, surprise, sadness, neutrality and anger. Our approach consists of creating an appropriate Convolutional Neural Network (CNN) architecture for emotion recognition, with accuracies of 96% in the IMDB gender dataset and 66% in the FER-2013 emotion dataset.

In addition, the team developed a real-time enabled guided back-propagation visualization technique, which displays the dynamics of the weight changes and evaluates the learned features.

6 Other relevant contributions

6.1 Recognition of Transparent Objects

Object recognition methods that rely on RGB-D sensors often fail with transparent objects such as bottles and glasses, which are common in domestic environments. To solve this problem, Hagg et al. [11] developed an extension of LINEMOD for detecting transparent objects, such that the method improves the recognition of transparent and semi-transparent objects without reducing the recognition rate of other objects significantly.

6.2 3D People Detection

The 3D people detection [12] uses a RGB-D camera which combines the advantages of LRFs (fast, accurate), monocular cameras (color information), and TOF cameras (3D information). The preliminary segmentation is based on a top-down/bottom-up technique which yields the capability of detecting partially occluded person, e.g. behind a desk or cupboard. The information gained from the local surface normals enable the system to detect a person in various poses and motions, i.e. sitting on other objects, bended to the front or side, walking fast/slow. As final machine learning technique, a Random Forest classifier is applied which outperformed the opponents AdaBoost and SVM. The presented approach is able to detect people (see also Figure 4) up to a distance of 5 meters with a detection rate of 87.29% for standing and 74.94% for sitting people.
Fig. 4: Detections for various pose configurations.

7 Open-Source Software

The current and former members of our team have developed various open-source components:

- an implementation of the execution models described by Mitrevski et al. [5] and an Unreal Engine simulation for reproducing some of the results presented there\(^3\)
- the motion detection framework described by Thoduka et al. [10]\(^4\)
- a real-time CNN-based face detection and emotion/gender classification model\(^5\)
- a ROS-based implementation of the tactile slip detector described by Sanchez et al. [7]\(^6\)

Our official GitHub page\(^7\) includes several other open-source repositories to which our current and past members have been contributing.

8 Conclusions and future work

In this paper the robot platform of the b-it-bots@Home team and its capabilities is presented. The integration of the functionalities in the robot platform is an ongoing process. In addition to that, some of our current research goals include lifelong learning, long-term experience acquisition, and skill generalisation, all of which are particularly important in the context of domestic robotics. The goals of our research group are reflected through four ongoing PhD projects, several master’s theses, and funded projects.

Other relevant R&D projects include communicating the robot’s intentions, large-scale and multi-floor mapping and autonomous exploration.

\(^3\) https://github.com/alex-mitrevski/delta-execution-models
\(^4\) https://github.com/sthoduka/fmt_motion_detection
\(^5\) https://github.com/oarriaga/face_classification
\(^6\) https://github.com/mas-group/tactile_slip_detector
\(^7\) https://github.com/mas-group
Acknowledgement

We gratefully acknowledge the continued support of the team by the b-it Bonn-Aachen International Center for Information Technology, the Bonn-Rhein-Sieg University of Applied Sciences and the Fraunhofer Institute for Manufacturing Engineering and Automation in Stuttgart for the help and support in providing us the Care-O-bot as a research platform for the RoboCup@Home competition.

References

Jenny (Care-o-bot 3)

The primary robot platform for the b-it-bots@Home team is the omni wheeled robot Care-O-bot 3. It is developed by the Fraunhofer Institute for Manufacturing Engineering and Automation (IPA) in Stuttgart, Germany. Specifications are as follows:

- **Base**: omni-directional platform, powered by 8 motors (2 motors per wheel: 1 for rotation axis, 1 for drive)
- **Torso**: 4-DOF
- **Arm**: Mounted on the base. KUKA LWR 7DOF with a three-finger Schunk SDH Gripper.
- **Tray**: sidemounted, can be moved up to transport objects.
- **Head**: contains a Microsoft LifeCam 1080p camera and an Asus Xtion Pro Live camera.

Also our robot incorporates the following devices:

- Manyears microphone array
- Microphone for speech recognition
- Two SICK S300 laser scanners
- One Hokuyo URG-04LX laser scanner
- SDH tactile sensors

Robot’s Software Description

For our robot we are using the following software:

- **Platform**: Robot Operating Systems (ROS) [13]
- **Navigation**: in-house ROS-based code
- **Emotion recognition**: In-house CNN models
- **Speech recognition**: Nuance VoCon
- **Object recognition**: A combination of approaches including CNNs, Ensemble of Shape Functions (ESF) and classical vision developed in-house
- **Arm control**: MoveIt! \( ^8 \).

\( ^8 \) moveit.ros.org