Birmingham Autonomous Robotics Club (BARC)
Motto: *Learning by doing*

Members: Lenka Mudrova, Sean Bastable, Maria Verghelet,
Fatma Faruq and Matei Vicovan-Hantascu
Email: robot-club@cs.bham.ac.uk
Affiliation: School of Computer Science, University of Birmingham, UK
Website: *http://www.birmingham-robotics.co.uk*

***Abstract*—Birmingham Autonomous Robot Club (BARC) connects students from the University of Birmingham, with a strong interest in robotics applications. This paper is part of our qualification for the Major Tournament of the ERL Service Robots (ERL-SR) challenge 2017-2018. Therefore, it provides an overview of our robot Dora and the developed software structure based on ROS middleware.**

It describes how this challenge relates to our interests and experiences and how we can achieve high reusability of our system by integrating different subsystems from other projects.

Finally, the conclusion summarises our motivation and relevance for this competition.

I. Introduction

Birmingham Autonomous Robotics Club (BARC) was established in 2011 in the School of Computer Science at the University of Birmingham. The main purpose was to provide an extra opportunity for students to gain additional knowledge about robotics and to work on real robotic platforms and projects. Several students involved in the team contributed to projects which were mainly used to promote robotics during the school's open days. For example a robotic waitress which accepted orders for drinks and delivered them.

In 2014, the structure of BARC was changed in order to incorporate the lessons learned and to allow the team to take part in robotics competitions. We participated successfully in Sick Days 2014 [1], RoCKIn@Home 2014 and 2015, and ERL-SR 2016-2017. At RoCKIn@Home 2014, we won two of the competition challenges: Getting to know my home and Welcoming Visitors. In 2015, we won Object Perception and we took 3rd overall place. This year, we would like to participate in ERL-SR as we still have many things to learn, improve and most importantly contribute.

The team has the support of the Intelligent Robotics Lab [2] in the School of Computer Science. The lab conducts research and has expertise in a variety of fields including but not limited to computer vision, manipulation, planning, architectures, reasoning and mobile robots. Furthermore, the lab has strong links with the industry. We are also starting to cooperate with other departments within our university, such as electrical and mechanical engineering.

This paper provides an overview of our system from last year and analyses the drawbacks of our solutions. Moreover, it extensively discusses what we are currently working on.

Fig. 1. Our logo represents our university campus with its famous clock tower, Old Joe.

II. Hardware

Our robot Dora is a Pioneer 3D-X differential robot platform with an in-house central supporting construction attached to it. This construction serves as support for the various sensors and the laptop which is the brain of the robot (see Fig 2). In terms of sensing Dora is equipped with 2 laser rangefinders and 1 depth camera. These sensors are primarily used to ensure safe navigation and people detection. Additionally, Dora is equipped with a microphone and speakers in order to enable Human-Robot-Interaction and to be able to participate in the tasks involving speech.

III. Software architecture

As our goal is to achieve high re-usability, modularity and openness, we built our software architecture on the Robot Operating System (ROS) middleware, using version Indigo [3] running on Ubuntu Trusty Tahr. Additionally, our system uses some of the publicly released packages from the STRANDS project [4]. This allows us to not only integrate the cutting edge research to our robot, but also to contribute to the evaluation of this research. Moreover, we integrate our own research where our interest overlaps with the focus of the competition. Our system architecture contains the following modules.

A. State machine

Even though AI planning is in our expertise, we use *finite state machines* to control and monitor the robot's states and actions. We have two main reasons for this decision. First,

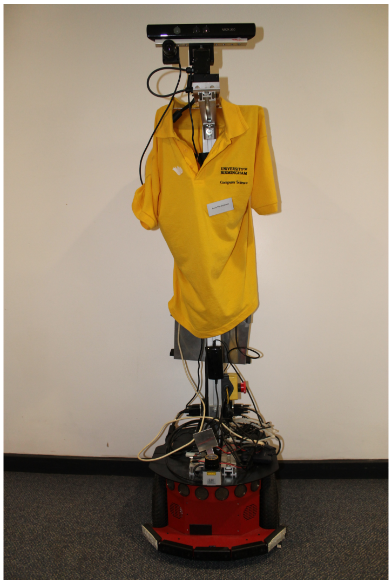

Fig. 2. Dora is an extended Pioneer 3D-X robot with sensors such as laser range finders, depth cameras and a laptop mounted on top.

the competition defines all *task benchmarks* as short scripts, so a robot does not have too much freedom in decisions on how to fulfil a task. Second, a state machine provides us with *repeatability* during testing.

For each of the benchmarks, we develop the state machine using ROS's SMACH package, that is "a task-level architecture for rapidly creating complex robot behaviour" [5]. All of the task's state machines are linked with the central state machine which communicates with the referee box and based on the accepted benchmarking test triggers specific state machines.

B. Navigation

We use standard ROS packages for localisation, mapping and low-level navigation, e.g. navigation planning a path from the initial coordinates to the goal coordinates. We observed that in some cases, such as passing narrow doors and passing through a doorway, a special robot behaviour would be better. Therefore, we are extending our system using high-level topological navigation [7] and we can strongly benefit from three main features:

- Waypoints and edges are managed in a database which allows even online modification.
- A special robot behaviour can be specified on the edges overriding standard move-base.
- A navigation policy [8] provides paths on the top of the topological map containing waypoints.

Thus, the robot must pass the surrounding area of the waypoint on the path. This can be mainly used to demand certain robot movements and keep the robot away from obstacles in an environment.

C. Mapping

is done by *OpenSlam's GMapping* algorithm [9] through the ROS wrapper package called slam gmapping. This approach uses a Rao-Blackwellized particle filter in which each particle carries an individual map of the environment. The particles are updated by taking into account both odometry and the latest observations from a laser range finder.

D. Localisation

in a known map uses an *Adaptive Monte Carlo Localization* (AMCL) [10] algorithm. This node is part of the ROS navigation stack package. It uses laser range finder readings to update a particle filter. Every particle represents a specific discrete state and stores the uncertainty of the robot being in that state/position within the known map. Also, every time the robot moves it uses the motor information to shift the particles to the corresponding direction.

E. Low-level navigation using move-base

was used last year as the only navigation system. It is a proven robust solution for domestic environments [11]. More specifically this node reads the odometry, the current pose estimate and the laser range finder scans and 3-D points clouds from a depth camera and safely drives the robot in the environment to a predefined goal. In order to navigate smoothly, it uses a combination of a global and a local planner. The global planner creates an optimal global path based on the robot's pose and a global cost-map. Then the local planner, which uses the *Dynamic Window Approach* algorithm [12], is responsible for following the global path and reactively avoiding obstacles.

F. Face detection and recognition

Face detection, see Fig. 3, is performed using the *Viola-Jones* algorithm [13]. The algorithm looks for faces by incrementally applying many simple Haar classifiers. The composition is performed by a cascade function, which needs to be trained *a priori* with many positive and negative images. The resulting classifier can find faces efficiently and independent of the size of the faces and light conditions.

Face recognition is performed by applying a *Local Binary Pattern Histogram* (LBPH) algorithm [14]. The principle of the algorithm is to build local binary patterns (LBP) for each pixel depending on a variable neighbourhood scheme. Then, it divides the formed LBP image into m local regions and computes the LBP distribution histogram from each region. Finally, classification is performed by comparison between the LBP histograms of a new face with the ones from the dataset.

G. Uniform Detection

Our uniform recognition uses colour segmentation of an input image in to two regions. A positive region corresponds to a colour within previously manually calibrated lower and upper bounds. In contrast, a negative region corresponds to the colours outside of these limits. Based on the size of the positive region, the person is classified as a Postman, Deliman or other.

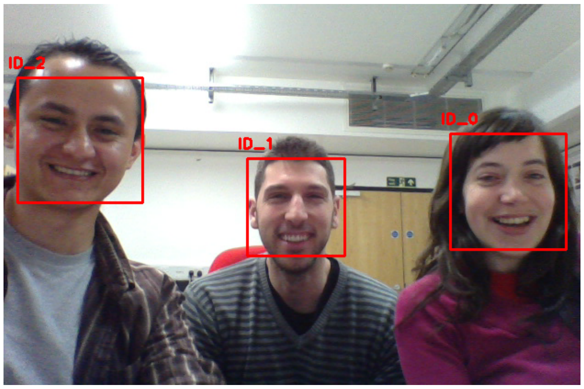

Fig. 3. An example of the face detection and face recognition algorithms. The red bounding boxes surround the successfully detected faces, while each of them is given a corresponding identification code.

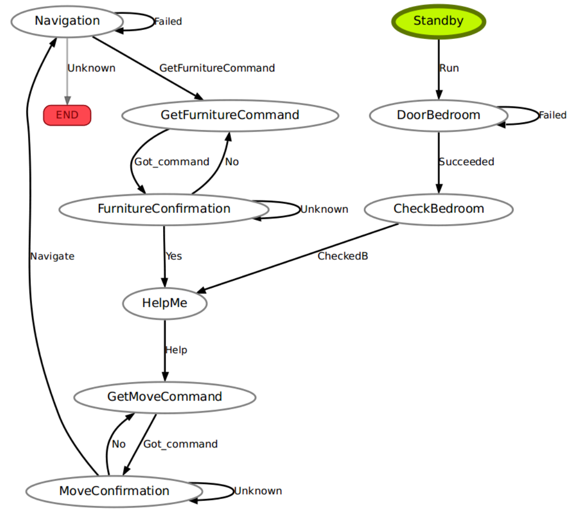

Fig. 4. The state machine for the task Getting to know my home

IV. Our focus and plan

This section summarises our intentions for our system this year. We would like to aim for high re-usability of code in our robotic system and to demonstrate that the current state of the art in AI algorithms mixed with our extensions can be successfully used to produce a robust, effective and complete robotic system with domestic applications.

A. *Getting to know my home*

In this task, a robot should recognise changes in a flat, such as open/closed doors, furniture which has been moved to another room or everyday objects which were moved. The robot can detect these changes autonomously or by cooperation with a human who uses speech or gestures. After detection, the robot is asked to use this new knowledge to execute some commands, such as to bring a mug on the dining table which is now in living room.

The state machine for this task from the RoCKIn@Home 2014 is in Fig. 4. Our system was built on two parts - autonomous detection of doors using the laser rangefinder and cooperation with a human in order to detect where the furniture is. In RoCKIn@Home 2015 we extended this system and we have now fully functioning system to accomplish this task.

B. *Welcoming visitors*

In this task, a robot should welcome visitors, recognise them and accompany them to a specific location in Granny's flat. Several components are needed in this task: computer vision for face recognition and for uniform detection. Moreover, machine learning techniques are needed to be able to learn these two patterns in order to recognise them. A robot can also benefit from speech recognition if face/uniform recognition does not provide a certain decision. Additionally, a robot must have robust navigation in an environment in order to accompany a person to a specific place. Finally, human-robot interaction is needed while accompanying a person.

We successfully completed this task during the RoCKIn@Home challenges. Our system was built on several components which are mentioned in Sec. III. Our core state machine is built for this task calling several components. We do not plan significant changes to our face recognition system, however, we will create a larger database of faces in order to improve the classification of the doctor.

Uniform recognition needs to be improved due to the following reasons. First, uniform detection was based only on colour segmentation of the whole image. This means that if there is a significantly large object of a certain colour belonging to a uniform (white/yellow) in the background, this object will be classified as the person. We previously tried to improve this by only looking for the uniform in the region of the image below a detected face, but the face detector was not robust enough in the presence of shadows. In addition, we plan to extend the uniform detection with a keypoint based image detector to search for the uniform logos. Both issues were limitations last year and we are improving them.

C. *Object perception*

We have won this task in RoCKIn@Home 2015 with an approach which uses transfer learning to recognise objects using a convolutional neural network (CNN) which has been pre-trained for another task, in our case, image classification [15]. An RGB-D pair of images is first pre-processed as shown in Fig. 5. Next, the images are processed by the CNN in order to extract a vector of image features for each RGB-D frame. This feature vector is used with a tree of classifiers and regressors to determine first the category of the object, then the object instance, and finally the pose of the object.

D. *Timetable*

We intend to participate in all benchmarks except for *Catering for Granny Annie's Comfort*. However, our priorities are the following. First, we will perform the complete *Getting to know my home*, *Welcoming Visitors* and *Object perception* benchmarks. We expect to participate in the *Visiting My Home* task benchmark and *Navigation* functional benchmark. We

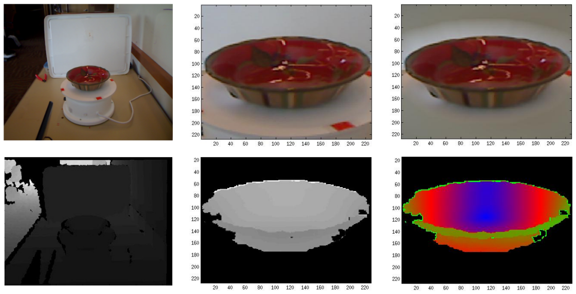

Fig. 5. The image pre-processing pipeline for the described algorithm. Top: The RGB image is first cropped down to size and then a fading algorithm is used to reduce the chance of unwanted features being detected in the background of the image. Bottom: The depth image is cropped in the same way and then coloured based on the depth data. This is required because the CNN is trained to detect features in RGB images and so we want the new image to describe the depth information in RGB. Image data used from the Washington RGB-D Object Dataset [16].

hope to participate in the *General Purpose Service Robot* and *Speech Recognition*.

V. Evaluation and benchmarking

In order to develop a robust system, we have decided to focus on intensive testing and evaluation. Therefore, we tried to find possible ways to build a test arena and equip it with real objects. However, this is impossible for us due to a strongly limited budget and lack of space. However, we came up with a better idea - we have decided to use one of our flats, see Fig. 6.

We have found such an environment which is similar to the arena with all of the tricky situations such as narrow corridors, door steps, a large mirror, etc. Therefore, we are able to test our algorithms in a realistic environment and improve their robustness.

Fig. 6. Long-term deployment of Dora in a flat

VI. Conclusion

Our team has strong relevance to domestic service robotics, as Lenka's research interests involve cooperation with humans and Sean's are in the area of computer vision. We would like to use our expertise to combine the state of the art in AI techniques with our own contributions. As a result, a robotic system will be created with high reusability, as we use ROS middleware.

We would like to achieve high robustness of our robotic system so that it can perform tasks repeatedly. In order to achieve this goal, we would like to put attention to verification and evaluation of our system.

We plan to cooperate more with first year bachelor's students in order to introduce them to robotics and provide them with some interesting challenges, which can be used during the competition. We believe that participating in the ERL-SR challenge will bring a lot of experience not only to the young students, but also to us. As a result, we are expecting to obtain more knowledge in the ongoing research in many fields of robotics.

References

[1] sick, *SICK robot day*, http://www.sick.com/group/de/home/pr/events/robot_day/seiten/robot_day_overview.aspx, accessed May 2015.

[2] IRLab, *Intelligent Robotics Lab (IRLab), School of Computer Science, University of Birmingham*, http://www.cs.bham.ac.uk/research/groupings/robotics/, accessed May 2014.

[3] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "Ros: an open-source robot operating system," in *IEEE International Conference on Robotics and Automation, workshop on open source software*, vol. 3, 2009.

[4] STRANDS, *STRANDS - Spatio-Temporal Represenations and Activities for Cognitive Control in Long-Term Scenarios*, http://strands-project.eu/, accessed May 2014.

[5] "SMACH - a task-level architecture for rapidly creating complex robot behaviour," http://wiki.ros.org/smach, accessed May 2014.

[6] robomongo, *Robomongo - Shell-centric cross-platform MongoDB management tool*, http://www.http://robomongo.org/, accessed May 2015.

[7] J. Pulido Fentanes, B. Lacerda, T. Krajník, N. Hawes, and M. Hanheide, "Now or later? predicting and maximising success of navigation actions from long-term experience," in *International Conference on Robotics and Automation (ICRA)*, 2015.

[8] B. Lacerda, D. Parker, and N. Hawes, "Optimal and dynamic planning for Markov decision processes with co-safe LTL specifications," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Chicago, Illinois, USA, September, 14–18 2014.

[9] G. Grisetti, C. Stachniss, and W. Burgard, "Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling," in *IEEE International Conference on Robotics and Automation*, 2005, pp. 2432 – 2437.

[10] D. Fox, "Kld-sampling: Adaptive particle filters," in *Advances in Neural Information Processing Systems (NIPS)*, 2001, pp. 713–720.

[11] E. Marder-Eppstein, E. Berger, T. Foote, B. Gerkey, and K. Konolige, "The office marathon: Robust navigation in an indoor office environment," 2010, pp. 300–307.

[12] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *Robotics Automation Magazine, IEEE*, vol. 4, no. 1, pp. 23–33, Mar 1997.

[13] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, 2001, pp. I–511–I–518 vol.1.

[14] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in *Computer Vision - ECCV 2004*, ser. Lecture Notes in Computer Science, T. Pajdla and J. Matas, Eds. Springer Berlin Heidelberg, 2004, vol. 3021, pp. 469–481.

[15] M. Schwarz, H. Schulz, and S. Behnke, "Rgb-d object recognition and pose estimation based on pre-trained convolutional neural network features," 2015.

[16] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view rgb-d object dataset," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1817–1824.