LCASTOR 2024 Team Description Paper*

Hariharan Arunachalam, Francesco Del Duchetto, Riccardo Polvara, Leonardo Guevara, Nikolaus Wagner, and Marc Hanheide

Lincoln Centre for Autonomous Systems (LCAS), University of Lincoln, Lincoln, UK
https://lcastor.blogs.lincoln.ac.uk/

Abstract. LCASTOR is the team from the Lincoln Centre for Autonomous Systems (LCAS) at the University of Lincoln, United Kingdom. It comprises academics, researchers and postgraduate students that have developed several solutions for the deployment of social robots in public environments. The main research objectives are focused on the integration between Deep Learning and Human-Robot Interaction (HRI) and, in particular, on robot learning, long-term autonomy, and HRI with untrained users. Our main motivations to participate in the RoboCup@Home Open Platform are 1) the adaptation of the solution developed by the research group in National and European projects to the RoboCup@Home tasks, and the dissemination and release of these solutions to the RoboCup@Home community, 2) starting a new robotics competition task force within LCAS to foster the involvement of undergraduate and postgraduate students in the field.

1 Introduction

LCASTOR (LCAS Team fOr RoboCup) team was formed in 2023 at the Lincoln Centre for Autonomous Systems (LCAS) for participation in RoboCup@HOME OPL competitions. The team was born building from the experience of the LCAS group in participating in past RoboCup@HOME SSPL competitions with the SPQReL team[1]. The main research objective of the team is to foster further collaboration between postgraduate students and academics for developing effective solutions for social and service robots in public spaces. LCAS gained a lot of experience in this field with the participation in several projects related to this topic. In particular, the recent projects STRANDs[2], Lindsey[3] and TARICS[4] have developed components for social robots in public environments. The main goals of the team are: i) integrate the individual outcomes of the involved projects into a more functional and robust social robot, ii) adopt these solutions for the RoboCup@Home environment (i.e., to solve specific tasks), iii) release

* Supported by School of Computer Science, University of Lincoln, UK.

[1] https://sites.google.com/dis.uniroma1.it/spqrel

[2] http://strands.acin.tuwien.ac.at/

[3] https://lcas.lincoln.ac.uk/wp/projects/lindsey-a-robot-tour-guide/

[4] https://tas.ac.uk/research-projects-2022-23/tarics/

and disseminate outcomes stemming from this initiative and the projects involved readily to the RoboCup@Home community, iv) establish a new robotic competition task force within our University to attract more students and foster collaboration between students and academics.

2 Scientific Contributions

In this section, we discuss the main recent scientific contributions achieved by our research groups and the research topics relevant to the RoboCup@Home competitions in general.

2.1 General objectives

Planning and plan execution. The deployment of robots in populated environments interacting with non-expert users requires facing many sources of uncertainty during task execution such as incomplete information about the environment or unpredictable behaviours coming from humans. Planning and plan execution under such uncertainties is also an important problem to be addressed within the RoboCup@Home competition and, in this context, we have recent research results.

In [14], we propose Next-Best-Sense (NBS), a decision-making framework that allows a mobile robot to explore an environment looking for objects while combining multiple criteria in a single utility function. Modelled following the traditional sense-plan-act paradigm, NBS iteratively select a new robot pose in order to efficiently explore an environment while carrying out a survey task. We further extend NBS in [13], where we propose a topological formulation of a particle filter for tracking multiple fruit harvesters in a polytunnel scenario, by integrating 2D LiDARs, RFID, and GPS readings.

Human-Robot Interaction. Social robots that are deployed in human-inhabited environments, like homes or public spaces, need to be able to perceive humans and plan behaviours that take into account their external and internal states. In this aspect, we are interested in improving the state of the art in user estimation and robot behavioural adaptation.

The long-term deployment, with a duration of more than three years to date, of a social robot in a public museum with the "Lindsey: the tour guide robot" project [2] has allowed us to study how people engage with robot technologies in such spaces and to consolidate our HRI technological developments. In [6], we present a fully integrated people perception framework, designed to run in real-time on a mobile robot. This framework employs detectors based on laser and RGB-D data and a tracking approach able to fuse multiple detectors using different versions of data association and Kalman filtering. To detect the users' engagement state during the interactions, we developed a learned regression model that can detect the users' group engagement level in real-time and from the robot's own camera [3]. Social robots also need to adapt what they do and say during interactions based on the detected users' state, for this reason, we designed a learning framework that adapts online, during the interactions, the robot behaviour to maximise the expected users' engagement [5,4].

The outcomes from the MeSAPro[5] project bring to the table additional features to our system which allow a simple bilateral Human-Robot Communication (HRC). The MeSAPro project developed a human-aware navigation framework that incorporates non-verbal communication based on body gesture recognition to communicate implicitly with the robot while using voice messages and visual alerts to let the human be aware of the robot's intentions [9]. This HRC was designed and evaluated for robot-assisted fruit harvesting operations but the gesture recognition approach based on the OpenPose is easily translated to the social robotics domain. Moreover, in [1] a causal framework has been exploited for modeling and predicting human spatial interactions in social robotics contexts.

Long-term autonomy. One of the main goals of the RoboCup@Home is to develop a system able to robustly navigate in dynamic environments subject to changes and unpredictable situations. In this context, [11] presented a localization and mapping system based on a spatiotemporal occupancy grid that explicitly represents the persistence and periodicity of the individual cells and can predict the probability of their occupancy in the future. The proposed representation improves the localisation accuracy and the efficiency of path planning. In [8], we present an approach for the topological navigation of service robots in dynamic indoor environments this approach uses a topological representation of the environment that simplifies the definition of navigation actions and is augmented with a spatiotemporal model that specifically represents changes that stem from events in the environment, which impact on the success probability of planned actions which allows the robot to predict action outcomes and to devise better navigation plans. In [10], we have also shown how better HRI can be facilitated by exploiting long-term spatiotemporal experience, similar to the approach above, but directly linking long-term autonomy with setting goals for a mobile robot. In populated environments, the ability to be able to predict the directions people are heading is useful for robots to plan suitable paths. The machine learning method in [15] allows us to learn a model for such predictions from long-term experience.

3 System Architecture and Implementation

The framework implemented for competing in @HOME competitions combines different home-made and third-party solutions for enabling different capabilities with the TIAGo robot. Figure 1 outlines the framework by showcasing the most important software components and how they communicate with sensors, actuators and with each other. The Annex section at the end of this document,

[5] `https://www.york.ac.uk/assuring-autonomy/demonstrators/robots-to-support-farming/`

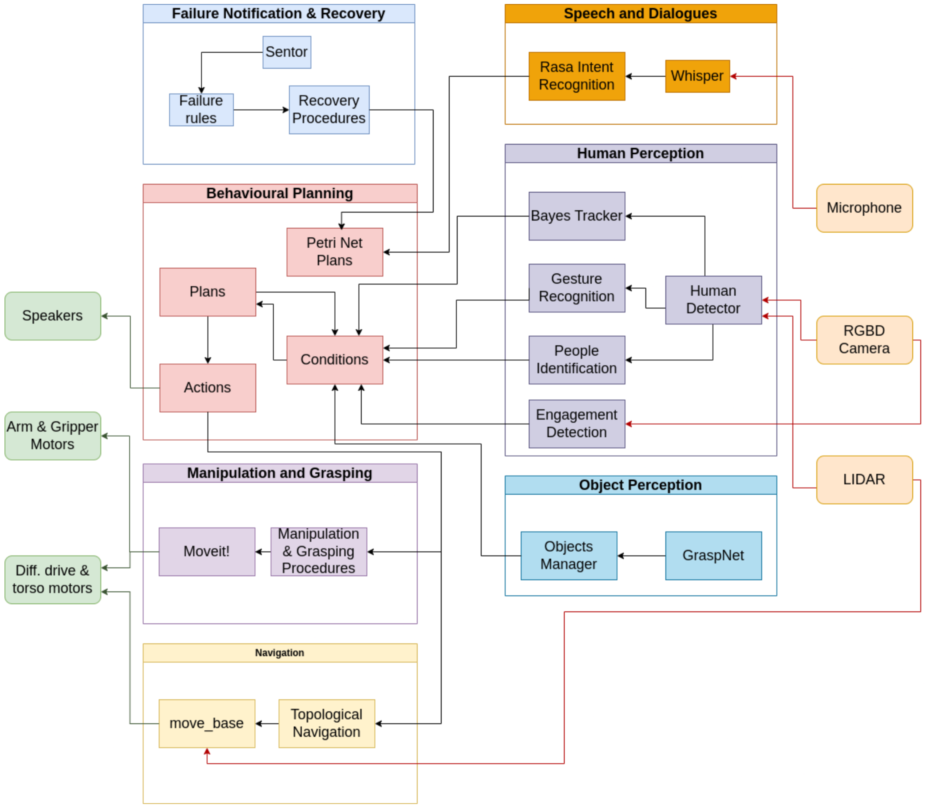

Fig. 1. Logical overview of the system architecture used to solve the competition tasks. The components developed by the LCASTOR team are highlighted in bold.

provides a comprehensive list of the publicly available repositories developed by us and a list of the third-party software.

3.1 Human Perception

For enabling social perceptual abilities, we need to make sure that the robot is able to perceive the humans in the environment, track their position over time, their pose and estimate their social state.

Bayes Tracker Filter. We developed an algorithm that is able to fuse the human detection from heterogeneous data (for example, the users leg detection or their upper-body detection) and provide a unique and consistent representation of them over time. By integrating multiple observations, the algorithm can associate them to specific user identities over time.

Engagement Detection. In order to give TIAGo the ability to understand whether the users are paying attention to the robot whilst it is speaking or performing a task, we have developed an engagement detector. The model, implemented in with the Keras[6] library, takes a temporal window of RGB camera images and provides a real-time assessment of the engagement level of the users in front of it. This information is fed into the robot plans so that it can perform contingent actions to recover the engagement of people or repeat the task when their are paying attention.

Gesture Recognition. Since non-verbal communication methods are useful in scenarios where people might have speech disorders, we implemented a gesture recognition methodology that allows TIAGo to interpret the user's body gestures as commands to perform different actions. The gesture recognition starts by extracting human body skeleton joints from RGB images using OpenPose[7]. Then, we construct a feature vector conformed by relative distances from each skeleton joint to the neck joint and the angles between consecutive joints. We normalize the feature vector and remove information from joints that are not crucial. We only use the upper body joints from the BODY 25 skeleton model which are transformed into a total of 28 features. Those features are used as inputs to a Random Forest classifier which was trained to classify 11 different body gestures These body gestures are then interpreted by TIAGo as commands to react in different ways.

People Identi cation. People identification is done with a encoder model, which produces a vector in the latent space at the output of the model. This model is an open-source implementation of Deepface [8]. Pretrained weights were used in the model. In the implementation, there are two phases namely learning and inference. The learning phase is done to record vector of any new face, and in the inference phase all recorded vectors are compared with new vector produced by the model to identify a person in front of the robot.

3.2 Navigation

The *Topological Navigation* module, is a high-level navigation framework that allows defining points-of-interest, also called nodes, on the metric map of an environment to enable robust navigation. Nodes in the topology are connected by edges which represent the actions that the robot can take to go from one node to the other. The framework is agnostic to the low-level navigation action used, however, in our competition's implementations we use the `move_base` software. Topological navigation enables tagging, with one or multiple tags, the various nodes defined in the environment to reflect location names or their properties. For example, for the competition tasks, we use tags to individuate locations in different rooms of a house and to indicate which robot actions are appropriate to be executed in specific locations. The topology and the node's tags are updated

[6] https://keras.io/
[7] https://github.com/CMU-Perceptual-Computing-Lab/openpose
[8] https://github.com/serengil/deepface

in real-time during execution by the robot based on the observations it takes in the environment.

In order to achieve a robust *obstacle avoidance* performance and safely plan a path among the people and the objects in the arena, we integrate observation coming from the 2d LiDAR mounted in the base of TIAGo together with the point clouds acquired from the 3d LiDAR located above the head of our robot and the one coming from the RGBD cameras located in the eyes. These three sensor data are integrated and mapped in the local cost map built around the robot footprint which is constantly updated to reflect the dynamic nature of the world in which the robot is moving. This solution proved to be efficient at identifying and avoiding even small obstacles located on the floor, such as soft drink cans.

3.3 Failure Noti cation and Recovery

The *sentor* package is a monitoring node that monitors the overall state of the ROS system of TIAGo. The package allows to define rules which represent the "normal", or "abnormal", state of other ROS nodes based on topic messages observations and to attach recovery procedures to any of implemented rule. Any time that a monitored situation happens,*sentor* will initiate the recovery procedure to return to a working state. This package enables us to handle exceptions and unexpected situations during the competition tasks execution to avoid stalling conditions.

3.4 Behavioural Planning

The execution of the robot behaviours is managed using the *Petri Net Plans* formalism, which allows to define conditional plans for the robot execution. The formalism was originally defined and implemented in [16], but extended by the LCASTOR team for easing the definition hierarchical plan by enabling a more practical reuse of plans as sub-plans, in the same way actions can be executed.

3.5 Speech and Dialogues

The robot's capability to understand voice commands is possible by using Whisper[9] as a speech-to-text converter. To mitigate the negative impact of noisy environments, an external microphone with an array of microphones that can be enabled/disabled is used. Only microphones pointing towards the front of the robot are enabled during human-robot interactions. Once the speech is captured, the intention recognition and entity name extraction are done by a Rasa-based chatbot model[10]. The Rasa model is trained to recognize specific intentions based on given sentence examples. If more than one entity name is required to be extracted from the speech, the rasa model is able to automatically generate a questionnaire that collects all the inquiries before sending them to the ROS planner for further action.

[9] `https://openai.com/research/whisper`

[10] `https://rasa.com/`

3.6 Object Perception

The perception, manipulation and general handling of objects are all tightly coupled together in our implementation. As input data for all this we use RGB-D data from an Azure Kinect camera [12], which provides high quality sensor outputs which are resistant to external disturbances due to the nature of the time-of-flight sensor used to acquire the depth images and the generally robust RGB sensor. We then feed these input data into an implementation of Grasp-Net [7] to simultaneously produce segmentation masks of the objects of interest, their full 6D-poses as well as high-confidence estimates of the ideal grasp pose for each object. All these annotations are handled by an Object Manager, which stores them and communicates them via custom ROS messages to other nodes requiring information about the location, pose or "graspability" of the objects of interest whenever the robot needs to interact with them.

To accommodate the increased computational demand required to run the implementation of GraspNet, we also have a more lightweight implementation of a simple Mask-RCNN running on standby, trained on the same object set as the GraspNet, which can be run as a more efficient substitute when grasping poses are not currently required. This setup allows for a much more efficient object perception framework, which delivers exactly what we require it to when we require it to, without unnecessarily wasting computational resources during downtimes.

4 Conclusions and Future Work

In this document, we have described the main scientific interests of the LCAS-TOR team members and the technical contribution that have been developed so far for participating in RoboCup@HOME competitions.

With the experience from the 2023 Competition, we plan to improve our framework making it more robust that can solve most @HOME tasks integrating navigation, vision, manipulation, speech and planning capabilities.

Participating in this year's competition will give us the possibility to show-case our current strengths and establish our team further for future local and international competitions.

References

1. Castri, L., Mghames, S., Hanheide, M., Bellotto, N.: Causal discovery of dynamic models for predicting human spatial interactions. In: Social Robotics. pp. 154–164. Springer Nature Switzerland, Cham (2022)
2. Del Duchetto, F., Baxter, P., Hanheide, M.: Lindsey the tour guide robot-usage patterns in a museum long-term deployment. In: IEEE International Conference on Robot & Human Interactive Communication (RO-MAN). IEEE (2019)
3. Del Duchetto, F., Baxter, P., Hanheide, M.: Are you still with me? continuous engagement assessment from a robot's point of view. Frontiers in Robotics and AI **7**(116) (2020)

4. Del Duchetto, F., Baxter, P., Hanheide, M.: Automatic assessment and learning of robot social abilities. In: Companion of the 2020 ACM/IEEE International Conference on Human-Robot Interaction. pp. 561–563 (2020)
5. Del Duchetto, F., Hanheide, M.: Learning on the job: Long-term behavioural adaptation in human-robot interactions. IEEE Robotics and Automation Letters **7**(3), 6934–6941 (2022)
6. Dondrup, C., Bellotto, N., Jovan, F., Hanheide, M., et al.: Real-time multisensor people tracking for human-robot spatial interaction (2015)
7. Fang, H.S., Wang, C., Gou, M., Lu, C.: Graspnet-1billion: A large-scale benchmark for general object grasping. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11444–11453 (2020)
8. Fentanes, J.P., Lacerda, B., Krajník, T., Hawes, N., Hanheide, M.: Now or later? predicting and maximising success of navigation actions from long-term experience. In: 2015 IEEE international conference on robotics and automation (ICRA). pp. 1112–1117. IEEE (2015)
9. Guevara, L., Hanheide, M., Parsons, S.: Implementation of a human-aware robot navigation module for cooperative soft-fruit harvesting operations. Journal of Field Robotics . https://doi.org/https://doi.org/10.1002/rob.22227
10. Hanheide, M., Hebesberger, D., Krajník, T.: The when, where, and how: An adaptive robotic info-terminal for care home residents. In: Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction. pp. 341–349 (2017)
11. Krajník, T., Fentanes, J.P., Hanheide, M., Duckett, T.: Persistent localization and life-long mapping in changing environments using the frequency map enhancement. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 4558–4563. IEEE (2016)
12. Microsoft Corporation: Azure Kinect DK Fact Sheet (2 2018), `https://news.microsoft.com/wp-content/uploads/prod/2019/06/Factsheet-Azure-Kinect-DK.pdf`, Rev. 1
13. Polvara, R., Del Duchetto, F., Neumann, G., Hanheide, M.: Navigate-and-seek: A robotics framework for people localization in agricultural environments. IEEE Robotics and Automation Letters **6**(4), 6577–6584 (2021). https://doi.org/10.1109/LRA.2021.3094557
14. Polvara, R., Fernandez-Carmona, M., Neumann, G., Hanheide, M.: Next-best-sense: A multi-criteria robotic exploration strategy for rfid tags discovery. IEEE Robotics and Automation Letters **5**(3), 4477–4484 (2020). https://doi.org/10.1109/LRA.2020.3001539
15. Sun, L., Yan, Z., Mellado, S.M., Hanheide, M., Duckett, T.: 3dof pedestrian trajectory prediction learned from long-term autonomous mobile robot deployment data. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). pp. 5942–5948. IEEE (2018)
16. Ziparo, V.A., Iocchi, L., Lima, P.U., Nardi, D., Palamara, P.F.: Petri net plans: A framework for collaboration and coordination in multi-robot systems. Autonomous Agents and Multi-Agent Systems **23**, 344–383 (2011)

Annex

Robot Hardware

The robotics platform used by the LCASTOR team for competing in the RoboCup@HOME OSPL competitions is a commercially available TIAGo robot developed by PAL Robotics[11]. The TIAGo configuration is shown in Figure 2.

External Computing Devices

Additionally, the computational requirements for executing the competition tasks are supported by an external Dell G17 laptop, which sits on the robot's laptop tray. The laptop is equipped with an intel i7 CPU and a NVIDIA GeForce RTX 2060 GPU. A Velodyne LIDAR is mounted on top of the robot with a metallic frame for holding the LIDAR above the robot.

Software Packages

The overall description of the software architecture in use for the competition is described in Section 3. The software components developed in-house by the L-CAS members of the team are:

- **Topological navigation**: `https://github.com/LCAS/topological_navigation/`
- **Bayes Tracker Filter**: `https://github.com/strands-project/strands_perception_people`
- **Engagement detection**: `https://github.com/LCAS/engagement_detector`
- **Gesture Recognition**: `https://github.com/LeonardoGuevara/mesapro`
- **Decision making and planning**: `https://github.com/LCAS/nbs`
- **Behaviour specification and planning**: `https://github.com/francescodelduchetto/PetriNetPlans`
- **Door opening**: `https://github.com/JakeSwin/Door-Handle-Detector-Ros`
- **Failure notification and recovery**: `https://github.com/LCAS/sentor`

The followings are the third-party software that we deploy on the robot:

- **OS**. Ubuntu 20.04
- **Middleware**. ROS noetic
- **Simulation**. Gazebo
- **Metric navigation**. move_base
- **Object detection**. YOLOv3
- **OCR**: Keras-OCR
- **Human pose detection**: OpenPose
- **Manipulation and grasping**. MoveIt!
- **Dialogue management**. Rasa
- **Speech recognition**. OpenAi Whisper

[11] `https://pal-robotics.com/robots/tiago/`