FBOT@Home LARC/CBR 2023 RoboCup@Home Team Description Paper

João F. S. S. Lemos Jardel D. S. Dyonisio Emilly G. Lamotte Luís F. M. Quadros Cristhian L. Froes
Marina Z. Rocha André P. Maurell Murilo C. S. Fonseca Pedro E. Nunes Cassio N. S. Teixeira
Gabriel A. Dorneles Murilo C. Bicho Aline S. Marchiori Luis F. F. Silva Igor P. Maurell
Jardel J. P. Pardo Nicolas F. Dias Henrique C. Bezerra Rodrigo S. Guerra Paulo L. J. Drews Jr

Abstract— **This team description paper (TDP) describes the team FBOT@Home and its Domestic Robotics Intelligent System (DoRIS), created by members of the NAUTEC group at the Universidade Federal do Rio Grande (FURG), Brazil. This document also describes our custom mechanical designs and public contributions we've made to the league, in the form of code packages developed by our team, such as: dataset generation unity, speech, behavior, world, simulation and more. All the developed modules are open-sourced and available at: `https://github.com/butia-bots`.**

I. INTRODUCTION

Domestic robots have been an ambition of engineers for decades. Furthermore, the growing challenges and pressures of modern life lead to a future where the use of robots in a domestic environment will be common. Thanks to current advances in technology, this aspiration is finally viable.

Pursuing that goal, we present the *Domestic Robotic Intelligent System - DoRIS*, a domestic robot designed by the FBOT@Home team. *DoRIS* is a a service robot consisting of a mobile platform, a torso (equipped with CPU and GPU units), an animatronic face and a manipulator. The team FBOT@Home is part of the NAUTEC research group. The group's goal is to research and develop robotics and automation solutions applied to real world problems.

Founded in 2001, NAUTEC is currently composed of 28 researchers with a PhD degree, 22 with a master's degree, 16 bachelor's and 10 undergrad students. The research group is leader in the field of analysis, modeling, perception, visualization and control of robots and devices, applied to oil, gas and energy industries, in addition to oceanic and coastal ecosystems, agriculture and education. FBOT@Home is one of four robotic competition teams, all grouped under FBOT project, through which NAUTEC fosters the development of team culture and technical skills of its members.

With DoRIS, the team FBOT@Home, previously known as ButiaBOTS, has already taken part in several RoboCup@Home competitions, securing notable achievements. We have achieved third place in the Brazilian Robotics Competition (CBR) for three consecutive years (2018 - 2019 - 2020), and attained the title of vice champion of CBR in 2021. Additionally, we secured third place in RoboCup Bangkok 2022 and ninth place in RoboCup Bordeaux 2023 (Open Platform League). Moreover, the team proudly holds the current champions' title of CBR 2022.

Fig. 1: DoRIS' architecture: external components

To introduce the robot, this document is organized as follows: Sec. II presents an overall description of DoRIS' architecture, Sec. III presents the contributions of our team, and Sec. IV presents the conclusion.

II. DORIS ARCHITECTURE

In this section, we present the architecture of DoRIS, showing the hardware that constitutes the robot and describing the software used.

A. Hardware

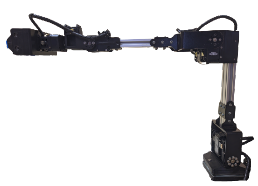
DoRIS, shown in Fig. 1, is composed of 6 major parts: (1) base, (2) processing units, (3) torso, (4) manipulator, (5) sensors and (6) head.

Base: A customized third generation PatrolBot is used as the deferentially driven mobile base. For robustness and safety of those around, the robot has a 0.8m tall triangular shaped torso with shelves for easy access to where all the internal components are installed.

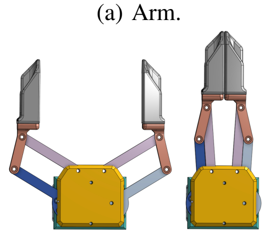
Processing units: There are two processing units: (1) an Intel NUC, used for general purpose and provided with an Intel Core i7 8705G, Radeon RX Vega M GL graphics and 8GB RAM, and (2) a NVIDIA Jetson TX2, dedicated for computing GPU specific tasks. To facilitate communication between both, a 2.4GHZ/5GHZ router is employed.

Sensors: Two ranging sensors are employed on the robot. A SICK LMS 100 rests on top of the mobile base, and is used for mapping, due its wide range and resolution. The other is a Hokuyo URG-04LX-UG01, closer to the ground, which is used to detect small obstacles during navigation. Additional sensors include a RGB-D camera (Intel Real Sense D435i), used for people and object detection and recognition, and a directional microphone (Rode VideoMic Pro) used for voice communication between humans and the robot.

Manipulator: The robot includes a self-built 5DOF arm, shown in Fig. 2a, employing Dynamixel MX-106T servo-actuators for the shoulder joints and Dynamixel MX-64T servo-actuators for the elbow, wrist and gripper joints. The structure of the links is mainly 3D printed in PETG polymer parts for the shoulder-elbow link and elbow-wrist, with some aluminum parts where a reinforced structure is required. The gripper, shown in Fig. 2b, is 3D printed in PLA and operates with a scissor-like motion, specifically designed to optimize both dimensions and movement precision.



(a) Arm.



(b) Gripper.

Fig. 2: DoRIS' manipulators

Head: One of the highlights of this robot design is its electro-mechanical 3D printed animatronic face. Manufactured with a combination of PLA, ABS and PETG, the head incorporates 13 hobby servo-actuators that control movements for the eyes, eyelids, eyebrows and jaw, making the face very expressive. Additionally 2 Dynamixel MX-64 servo-actuators are used for the neck, controlling pan and tilt. Facial expressions are controlled by the main CPU, which processes the respective ROS nodes, sending the appropriate commands to the servo-actuators through an Arduino UNO. Currently there are five pre-programmed standard expressions: happy, sad, neutral, angry and scared. Beside these, there is a thread that triggers a routine of blinking movements to make the robot look more natural. The face is used to associate emotions to DoRIS' progress in ongoing tasks. For example, if the robot succeeds accomplishing a task, it makes a happy face, whereas if something is uncomprehended it changes to a sad expression. The robot's jaw also moves accordingly to the speech being made. As the model in use became obsolete, it was decided to redesing DoRIS' face, Fig. 3 shows the new design elaborated with Onshape®. The current iteration is more robust than the previous one while also being manufactured through 3D printing technology.

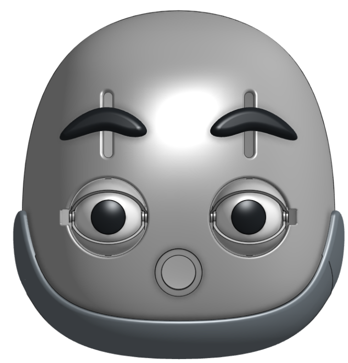


Fig. 3: DoRIS' new face design

B. Software

The diagram displayed in Fig. 4 presents a simplified version of DoRIS' system. The Robot Operating System (ROS) is used as middleware to connect all the packages and make them able to work together. The core of our system is the Behavior module, that uses ROS SMACH [1] to execute each task as a robust state machine.

A complete explanation of the software architecture, including the other modules mentioned in the diagram, will be more deeply explained next:

World: The World module has plugins to connect a key-value Redis Database [14] with services, topics or parameters from ROS. Each plugin has its own functions, not only to write and read data from database but to filter and transform it, if needed. Furthermore, the plugins are able to open services or topics to provide the Behavior module with access to memorized data. In short, the database is the robot's long term memory and the plugins are the filtering interface to connect the memory with the rest of the system.

Vision: The vision system of the team's robot is composed of several packages for each of the following tasks: object recognition, people tracking, face recognition and segmentation. Object recognition employs YOLOv8 [12]. People tracking utilizes DeepSORT [11] to track people detected by the object recognition package. For face recognition, the system adopts the pipeline proposed by OpenFace [8], modifying the detection and classification steps. Additionally, there is an extra package called Image to World, which is used to estimate the three-dimensional oriented bounding box of each recognition.

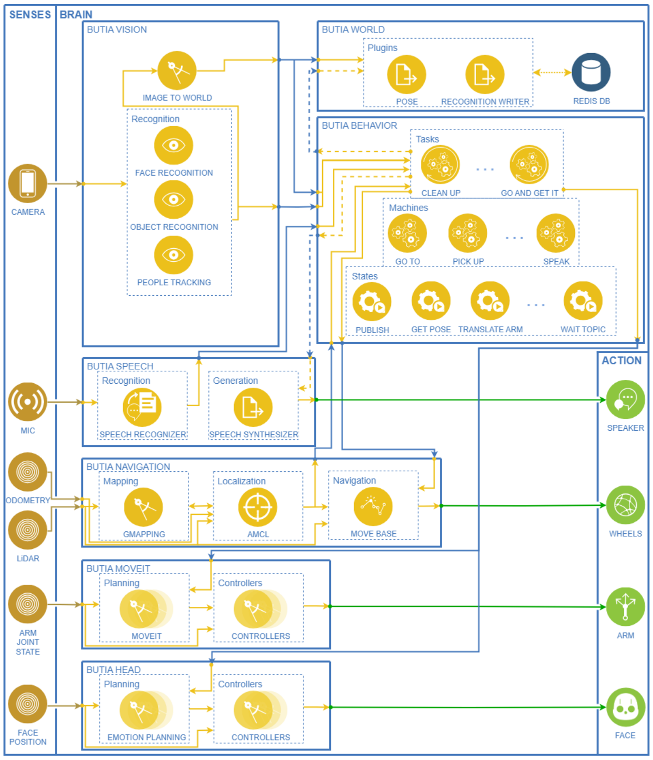


Fig. 4: Diagram of DoRIS' software architecture. Communications done by pointed lines are requests and by normal lines are message passing.

Speech: To perform speech recognition, the Whisper model[13], a large pre-trained model, is utilized. Furthermore, to synthesize DoRIS' voice, ESPnet2 is employed [15] with the *kan-bayashi/ljspeech_vits* model. Both modules are used offline to achieve better results in a competition environment. ESPnet2 also enables the generation of DoRIS' own voice. Currently, the package butia_speech is used for several tasks, such as: take out the garbage, carry my luggage, recepcionist and take orders. This allows the robot to have a complete conversion between speech-to-text and text-to-speech. For the GPSR task, natural language understanding and generation is implemented by using the langchain[7] software framework for developing applications powered by large language models.

Navigation: To be able to navigate and localize itself, the robot employs a Simultaneous Localization and Mapping (SLAM) system, which builds a map of the environment where the robot is. The SLAM system is based on a grid representation, along with Rao-Blackwellized Particle Filter(RBPF) [10]. This method comes along with the ROS package *gmapping*[2]. The localization is done by using a Monte Carlo Localization (MCL) method with adaptive sampling of particles, provided by the *amcl* package[3]. With a map and knowing its location, the robot is able to navigate throughout the environment. The navigation system works based on cost maps, dividing the environment into a grid and giving each cell a value of occupancy. A global cost map is build upon the environment map provided by the SLAM and a local cost map is generated with the information from the global cost map and from the sensors. Planners are employed to choose the best path through the cost map, with the global planner being a Dijkstra algorithm and the local planner a classic trajectory planning algorithm together with a Dynamic Window Approach (DWA) [9], both available on the ROS package *move_base*[4].

Manipulation: The robot utilizes the MoveIt[1] motion planning framework. This package supports several motion planning and inverse kinematics backends, such as OMPL[1] for motion planning and Orocos KDL[1] for inverse kinematics, which are the solutions employed by the team for the manipulation component of our software architecture. In order to handle the composition of several complex atomic manipulation tasks into simpler ones, the ROS SMACH state machine library is utilized for sequential and parallel execution of these atomic tasks.

III. CONTRIBUTIONS

The team has made significant contributions to the field of robotics through research, development, and the organization of events aimed at fostering and inspiring the study of robotics.

Brazilian technical committee @Home: The team has representation on the technical committee of the league through one of its members. In this role, responsibilities include contributing to the development and implementation of the rule book for tasks, assisting in the organization of the league during the event, and evaluating the materials submitted by teams for classification in the Brazilian Robotics Competition.

Documented repositories:

- Butia quiz [5]: This package provides the ability for the robot DoRIS to answer previously known questions.
- Butia face recognition [4]: Implementation of age, gender, and region detection with or without mask usage.
- Butia speech [6]: This package provides some tools to allow the robot to speak and listen.
- Butia Dataset Generator [3]: Unity Editor project for the synthetic computer vision dataset generation tools used by our team.

An Open-Source Robot and Framework for Research in Human-Robot Social Interaction: Human-Robot Interaction (HRI) is essential to the widespread use of robots in daily life. Robots will eventually be able to carry out a variety of duties in human civilization through effective social interaction. Creating straightforward and understandable interfaces to engage with robots as they start to proliferate in the personal workspace is essential. Typically, interactions with

[2]ROS Wiki: http://wiki.ros.org/gmapping
[3]ROS Wiki: http://wiki.ros.org/amcl
[4]ROS Wiki: http://wiki.ros.org/move_base
[1]Official Website: https://moveit.ros.org/
[1]Official Website: https://ompl.kavrakilab.org/
[1]Official Website: https://www.orocos.org/kdl.html

simulated robots are displayed on screens. A more appealing alternative is virtual reality (VR), which gives visual cues more like those seen in the real world. In this study, it is introduced Jubileo, a robotic animatronic face with various tools for research and application development in human-robot social interaction field. Jubileo project [2] offers more than just a fully functional open-source physical robot; it also gives a comprehensive framework to operate with a VR interface, enabling an immersive environment for HRI application tests and noticeably better deployment speed.

RoboWeek 2018: The robotics event, created by the team at Universidade Federal do Rio Grande (FURG) in collaboration with the Southern Brazilian chapter of IEEE-RAS and the Robocup Federation, was specifically designed to popularize robotics. The main goal of this event was to introduce both fundamental concepts and state-of-the-art developments in the field of robotics to university students and professionals. In addition, the event was focused on practical activities in groups, and at its conclusion, outstanding demonstrations were recognized and awarded.

Brazilian Robotics Competition 2019: The Brazilian Robotics Competition (CBR) is the largest university robotics competition in Brazil, held in partnership with the Institute of Electrical and Electronic Engineers (IEEE). In 2019, FURG had the honor of hosting the Brazilian Robotics Competition (CBR), with the team playing a pivotal role in the event's organization and execution. This involvement further established the institution as a leader in the field of robotics.

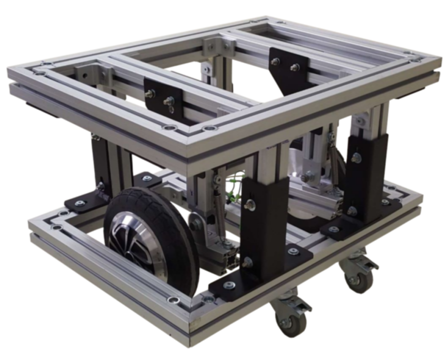


Fig. 5: Mobile base currently in development.

Low cost mobile base: The team is currently developing an affordable, high-capacity standard DIY robot base platform capable of carrying payloads up to 120kg, shown in Fig. 5. This initiative aims to provide a cost-effective solution to support @Home teams, facilitating a smooth and successful launch of their projects. In pursuit of this goal, we are repurposing the hardware found in self-balancing scooters, colloquially known as hoverboards. The adoption of this approach not only reduces costs but also ensures a robust and reliable system thanks to the well-established and proven components. The mechanical design will be developed around widely available hardware parts such as the standard square cross-section extruded aluminum bars and connection accessories, industrial caster wheels, springs and other supporting structures.

IV. CONCLUSIONS

This paper outlines the approaches used by the FBOT@Home team for the RoboCup@Home competition. It provides an overview of the robot DoRIS, encompassing its hardware composition including the base, processing units, torso, manipulator, sensors and head. The software structure is described with details of its implementations, demonstrating the use of a World module to provide a detailed portrait of the environment facilitating the decision making. Furthermore, we present the significant contributions of our work, demonstrating its practical applicability in real-world scenarios.

REFERENCES

[1] Jonathan Bohren and Steve Cousins. The smach high-level executive [ros news]. *IEEE Robotics Automation Magazine*, 17(4):18–20, 2010.

[2] Jair A. Bottega, Victor A. Kich, Alisson H. Kolling, Jardel D. S. Dyonisio, Pedro L. Corçaque, Rodrigo da S. Guerra, and Daniel F. T. Gamarra. Jubileo: An open-source robot and framework for research in human-robot social interaction, 2022.

[3] ButiaBots. Butia dataset generator. `https://github.com/butia-bots/butia_dataset_generation_unity`, 2020.

[4] ButiaBots. Butia face recognition package. `https://github.com/butia-bots/FaceScanner/`, 2022.

[5] ButiaBots. Butia quiz package. `https://github.com/butia-bots/butia_quiz/`, 2022.

[6] ButiaBots. Butia speech package. `https://github.com/butia-bots/butia_speech/`, 2022.

[7] Harrison Chase. LangChain, October 2022.

[8] Brandon Amos et al. Openface: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science, 2016.

[9] Dieter Fox et al. The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1):23–33, 1997.

[10] Giorgio Grisetti et al. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE transactions on Robotics*, 23(1):34–46, 2007.

[11] Nicolai Wojke et al. Deep cosine metric learning for person re-identification. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 748–756. IEEE, 2018.

[12] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. YOLO by Ultralytics, January 2023.

[13] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR, 2023.

[14] Salvatore Sanfilippo and Pieter Noordhuis. Redis, 2009.

[15] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai. ESPnet: End-to-end speech processing toolkit. In *Proceedings of Interspeech*, pages 2207–2211, 2018.