ACSO BahiaRT@Home 2023: Team Description Paper RoboCup@Home League*
Web Site: http://acso.uneb.br/bill/

Tamir de Almeida Amorim[1] Reinan Carvalho Amaral[1] João Paulo S. de Santana[1] Cauan Max C.F. Reis [1]
Davi Miguel Brito Barbosa[1] Josemar Rodrigues de Souza[1] Robson Marinho da Silva[1] Marco A C Simões[1]
Jorge Campos[1] Ana Patricia F. Magalhães Mascarenhas[1]

Abstract— **This paper presents the ACSO Bahia Robotics Team (BahiaRT) and describes an autonomous service robot named Bot Intelligent Large capacity Low cost (Bill) and its functions, such as navigation, manipulation, people and object recognition, human-robot interaction and decision making as well as the Bill's hardware and software systems. Furthermore, the paper highlights research interests and scientific contributions of BahiaRT from the Center of Computer Architecture and Operating Systems (ACSO) at the Bahia State University (UNEB).**

Keywords: Bill, service robotics, object detection, assistive robotics, speech recognition, RoboCup@Home.

I. Introduction

The Center of Computer Architecture, Intelligent Systems and Robotics (ACSO) at the State University of Bahia (Uneb) has been participating in RoboCup with the BahiaRT team since 2009 in leagues such as 2D Soccer Simulation, Mixed Reality, 3D Soccer Simulation, and @home, among others.

The service robot proposal of BahiaRT for the RoboCup@Home league is called Bill (Bot Intelligent Large capacity Low cost). It was born in 2014 as a result of research projects in assistive robotics. Its main functions are communicating with humans through natural language processing, recognizing objects and faces, and navigating through unknown environments. BahiaRT with Bill got 13th place in 2015 RoboCup@Home and 21st place in 2016. Specifically, in RoboCup@Home Brazil, got second place in 2015, 2016 and third place in 2017.

Over the years Bill has been improved to better meet the @home challenges (Figure 1). This paper presents the third generation of Bill, named **Bill Estranho**, and its main improvements. This year, a great part of the BahiaRT effort was dedicated to building Bill Estranho. This work involved re-engineering the hardware using new components, redefining the architecture, and changing the operating system to ROS2. As a consequence, all the other functionalities, i.e. voice recognition, face and object recognition, navigation, and manipulation needed to be updated.

Section II describes the main advances and scientific contributions of BahiaRT to assistive robotics. Section III introduces Bill and Section IV describes its architecture and main functionalities. Section V presents the conclusions and future work. Finally, at the end of this Team Description Paper (TDP) there is a brief description of each hardware and software component used in Bill's development.

* This work is partially funded by CNPq/PIBIC, FAPESB/IC and UNEB/PICIN

[1]Centro de Pesquisa e Arquitetura de Computadores, Sistemas Inteligentes e Robótica (ACSO), Universidade do Estado da Bahia (UNEB), Salvador, BA, Brazil. teambahiart@gmail.com

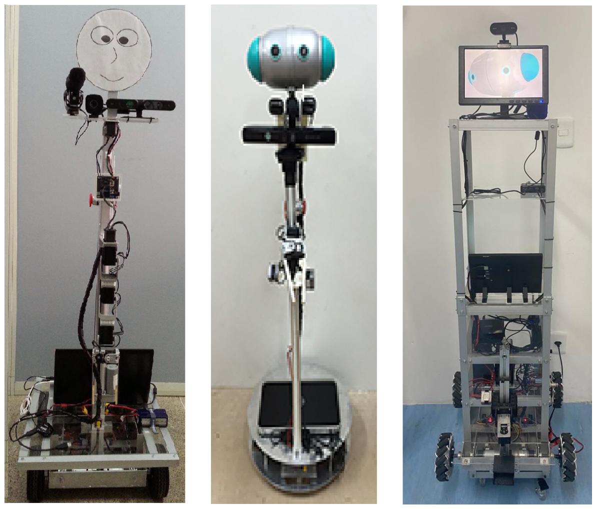

Fig. 1. Bill's evolution over the years.

II. BahiaRT's advances in innovative technology and research interests

The main research interests of BahiaRT involve robotics and artificial intelligence, specifically focusing on interaction and cooperation between human-robot, applications of artificial intelligence for services, standardization, and system integration, in addition to the construction of intelligent robots with low-cost components and open source. Developing an autonomous robot like Bill requires knowledge of computer and mechatronics technology to integrate systems and hardware solutions.

Bill uses various innovative technologies in an integrated approach, such as Robot Operating System (ROS) and its packages, Arduino boards, kinect OpenNI library, TurtleBot arm, computational vision, and speech algorithms, among others.

The experiments and texts used to evaluate Bill's behaviors involve specifications of parameters, control tasks, details of the software and hardware, innovative technologies, and adoption of other challenger tasks in Artificial Intelligence (AI).

However, despite the importance of this type of research, there are still not several autonomous robots running in human daily lives. Furthermore, as well as in other sciences, autonomous robots can achieve different results in practice due to the occurrence of unpredictable events and environmental changes. Therefore, there is still a long road ahead, with much work to be done for spreading services robots in residences around the world.

To fill this gap, practitioners should provide feedback on robotic research, sharing experiences for reducing differences among countries and their laboratories. To allow replication of research and to contribute to the development of robotic research, BahiaRT will provide access to the codes, hardware, and software list in this paper and on Bill's website.

III. Bill Estranho

Based on the characteristics of robots built by ACSO, used in competition over the last 17 years, and observations on equipment from other parts of the world, we present Bill, base type 3, code name Estranho, whose main feature is 04 differential wheels (Mecanum Wheels) (Figure 2).

In general, Bill's body was built of aluminum and covered with clear acrylic sheets. It contains the necessary hardware to control the robot. The base of the body is attached to 4 differential wheels (Mecanum Wheels) responsible for locomotion. On top of the body, we have a monitor, a video camera, and a sound box, which represent Bill's head and help the robot's vision and interaction with the outside world. Bill also contains a mechanical arm, attached to the front of the body, designed to allow object manipulation.

In Bill Estranho the electronics were improved with better protection for short circuits and higher power. Use of ROS2 [1], micro-ROS (puts ROS 2 onto microcontrollers) [2], ESP32 [3], etc. And in an unprecedented way, the use of the concept of Fault Tolerance [4] Fox/Dog method [reference: author] in Service Robots.

IV. Bill's Architecture

The Bill's architecture (Figure 3) is basically divided into levels: a high level, containing the functions related to the robot's abilities; and the low level, containing the controllers and drivers that send signals and receive commands from the sensors.

As shown in Figure (Figure 3), at the top of the architecture (in orange), is the interaction layer with the external world. Bill contains two monitors, one representing Bill's face (attached to its head) and another used for input and output via console (attached to his body). Also contains a microphone and speakers, Lidar, Kinect, and webcam. Manual control can also be performed via a joystick using a mobile application.

Bill contains a gateway and a computer as part of his body. All the access to him for installing new software is performed remotely from external workstations using VNC software.

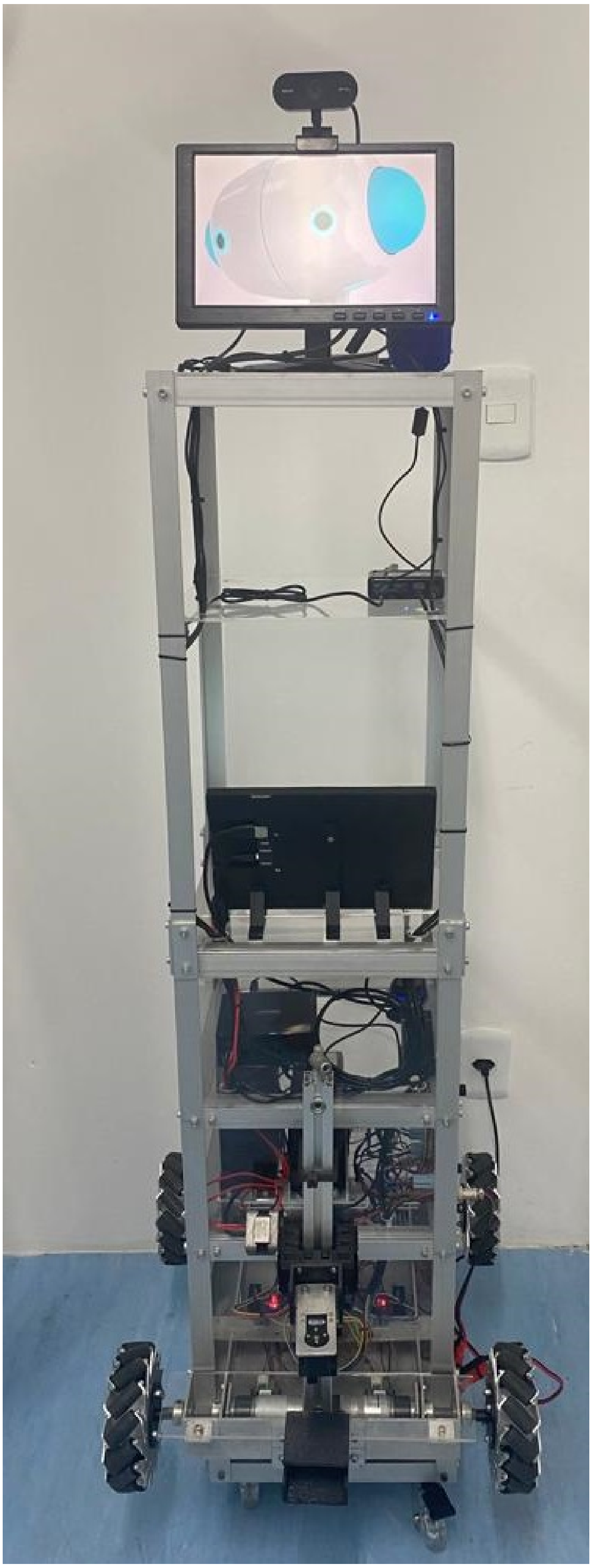

Fig. 2. Bill Estranho.

The fault-tolerance Fox-Dog method is implemented using two ESP32 microcontrollers replicated (in red in Figure 3) with ROS. They control the left/right motor front (LMF/RMF) and the left/right motor rear (LMR/RMR). So, if one microcontroller fails, the other takes over.

Bill also contains a control panel with LCD display that allows monitoring voltage, temperature, and connection, generating alarms in dangerous situations.

In addition, two batteries provide energy.

At the end of this paper, there is an Addendum (Bill Hardware and Software Description) with the characteristics of the Bill's components.

A. Motion

Bill's locomotion system consists of 4 Mecanum wheels, which allow it to move in any direction and turn around its own axis with ease. Mecanum wheels are made up of a series of diagonally oriented rollers. Each roller has a specific orientation, resulting in an "X" configuration when viewed from above. This unique wheel configuration allows the robot to move laterally, diagonally, and even in circles.

The four wheels are numbered starting from the front of the robot and working from left to right (Figure 4). For the motion, the linear velocities in the main directions (Vx and Vy) and the angular velocity around the Z axis (W) are converted into angular velocities for each wheel (Wi), computed according to the following equations [5]:

W0=1/r (Vx - Vy – (a-b)*W)
W1=1/r (Vx + Vy + (a+b)*W)
W2=1/r (Vx + Vy – (a+b)*W)
W3=1/r (Vx -Vy + (a+b)*W)

B. Navigation

Navigation is the keystone for efficient execution and environment interaction for robots. The components used by Bill for navigation are: encoders output, odometry, gmapping (ROS), move_base (ROS), Adaptive Monte Carlo Localization (AMCL) (ROS), map_server (ROS) and 360° laser scanner. The encoder data is used by odometry module to estimate the movements of the robot in space. Further, the odometry data is used to trace trajectory to a desired target by the move_base. Once all data are published, the simultaneous mapping and localization using the AMCL [6] is activated integrating the 360° laser scan data. Simultaneous Localization And Mapping (SLAM) approach is used to map the environment and provide self-localization in this map. First, the map is built using the incremental mapping package, Hector Mapping [7]. Then, the grid map generated is done by RPLIDAR 360 Laser Scanner sensor, which is able to get the 2D world data. The next step is creating the path planning based on the occupancy grid map that is updated based on the Dynamic Voronoi ROS package approach [8]. Then the shortest path to the goal is computed by means of the D* Lite algorithm [9] to ensure obstacle avoidance over incremental mapping. The motion planning in charge of getting the path planning and relating linear and angular motion is triggered, which applies the kinematics control law, and sends a message to low-level control.

C. Object Manipulation

Object manipulation plays an important role to interact with a home environment. To meet that requirement, Bill has an arm based on the TurtleBot Arm [10], composed of 5 Degrees of Freedom (DoF) including a gripper, which will allow the robot to grab lightweight objects. This entire system is controlled by an Arduino Arbotix-M [11], a model designed to work with servomotors. Several tests were carried out simulating trajectories that the arm must move, with satisfactory results. We integrated the control software developed in ROS 2 with the electronic components of the arm. We performed extensive tests to adjust and calibrate the control parameters, ensuring that the manipulator arm worked correctly. In addition, we explore the Gazebo [12] simulation environment. The next step was to transfer the knowledge acquired in the simulator to Bill's physical arm.

D. Vision

This module is responsible for receiving, processing, and responding to external stimulus from image capture. It is divided into two sub-modules which are facial recognition and object recognition.

The object detection in Bill utilizes the YOLO (You Only Look Once) version 5 [13]. YOLO is state-of-the-art in object recognition, capable of real-time detection of a wide range of objects. The training of Bill was conducted considering 200 objects with 30 photos each, using different locations and positions. This number was chosen to perform a diverse test. We employed varying numbers of objects in each image, always aiming to analyze the impact of parameters and the number of photos used in the recognition training.

Facial recognition has two main stages: facial detection and facial recognition. Following Bill's philosophy (high capacity at low cost), algorithms were evaluated to find a good trade-off between accuracy and response time. In the facial detection phase, we used HaarCascade [14], a technique for detecting Haar features, which are simple rectangular patterns used to identify specific features in an image. This allows us to locate and isolate the face from the rest of the image. Next, we employed the Dlib algorithm [15], which utilizes landmark detection technique to identify 68 coordinate points on the face, mapping and extracting this information for training. Both techniques are available in the open-source OpenCV library. Recognition tests were conducted with 55 students from UNEB (State University of Bahia). After a brief training, the algorithm was capable of recognizing each student individually and was also able to identify specific individuals within a group.

E. Speech Recognition and Voice

The voice is the most used form of human-machine interaction to give commands to the robot, either through the line of command or natural language. Due to Bill's upgrade to ROS2, we are currently working on the adoption of Google Speech Recognition software to enable voice interaction.

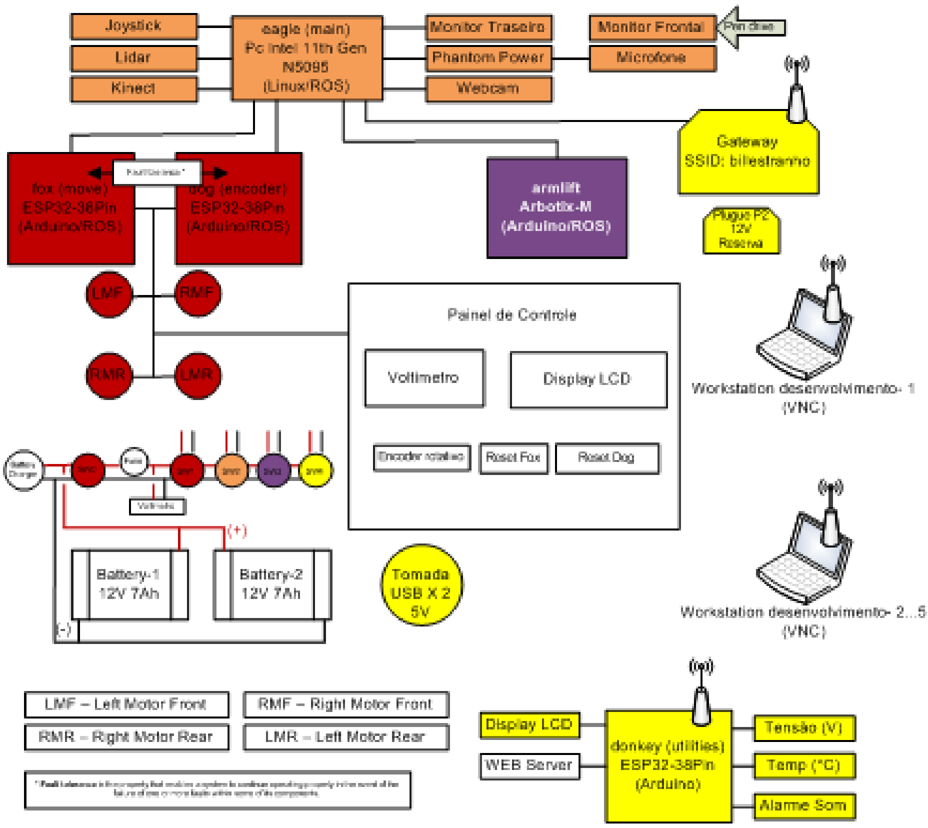

Fig. 3. Bill Estranho Architecture.

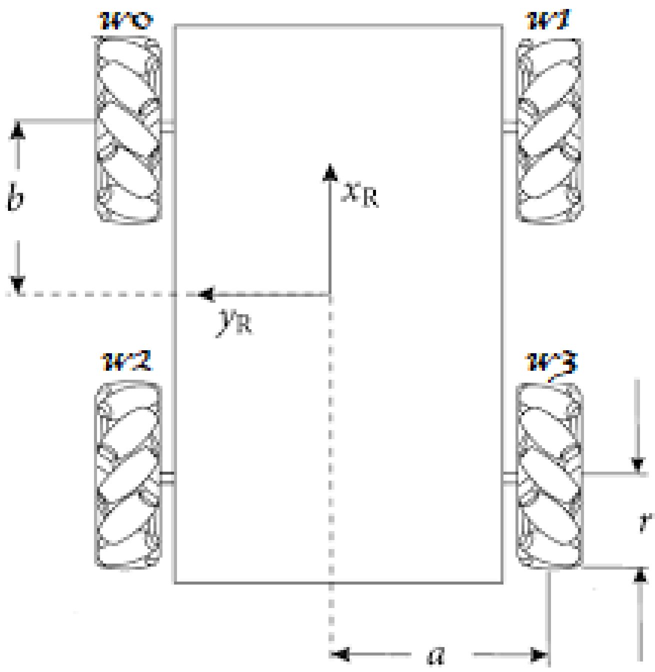

Fig. 4. Wheels and base's configurations.

V. Conclusions and Future Work

Although we still don't have metrics to validate Bill Estranho's performance, we can mention that the improvements observed are notorious; especially those related to the use of ROS2 and micro-ROS due to the ease and interaction between the various parts of the robot. Another highlight was the results obtained with the use of Esp32, a robust and versatile microcontroller that certainly "came to stay" So far, we are satisfied with what has been observed and intend to continue investing in this new hardware and software architecture.

References

[1] "Ros2," https://docs.ros.org/en/foxy/index.html, accessed: 2023-07-21.

[2] "micro-ros," https://micro.ros.org/, accessed: 2023-07-21.

[3] "Esp32," https://www.espressif.com/en/products/socs/esp32, accessed: 2023-07-21.

[4] B. W. Johnson, "Design and analysis of fault-tolerant systems for industrial applications," in *Fehlertolerierende Rechensysteme / Fault-tolerant Computing Systems*, W. Görke and H. Sörensen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1989, pp. 57–73.

[5] H. Taheri and N. Qiao, Bing Ghaeminezhad, "Kinematic model of four mecanum wheeled mobile robot," *International Journal of Computer Applications*, vol. 113, no. 3, pp. 6–9, 2015.

[6] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, vol. 2. IEEE, 1999, pp. 1322–1328.

[7] S. Kohlbrecher, O. Von Stryk, J. Meyer, and U. Klingauf, "A flexible and scalable slam system with full 3d motion estimation," in *Safety, Security, and Rescue Robotics (SSRR), 2011 IEEE International Symposium on*. IEEE, 2011, pp. 155–160.
[8] B. Lau, C. Sprunk, and W. Burgard, "Improved updating of euclidean distance maps and voronoi diagrams," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 281–286.
[9] J. Neufeld, M. Sokolsky, J. Roberts, A. Milstein, S. Walsh, and M. Bowling, "Autonomous geocaching: Navigation and goal finding in outdoor domains," in *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 2008, pp. 47–54.
[10] "Turtlebot3 documentation," https://emanual.robotis.com/docs/en/platform/turtlebot3/overview/, accessed: 2023-06-29.
[11] "Arbotix robot controller," https://www.trossenrobotics.com/p/arbotix-robot-controller.aspx, accessed: 2023-07-21.
[12] "Gazebo: 3d robot simulation made easy," http://gazebosim.org/, accessed: 2023-06-29.
[13] "Yolo," https://https://pjreddie.com/darknet/yolo/\/yolo\/, accessed: 2020-08-23.
[14] "Haar," https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html, accessed: 2023-07-21.
[15] "Dlib algorithm," https://medium.com/brasil-ai/mapeamento-facial-landmarks-com-dlib-python-3a200bb35b87, accessed: 2023-07-21.

Bill Hardware and Software Description

To provide completely autonomous operation, Bill owns two main modules of control: (i) the High-level control, which includes algorithms to solve functionalities such as global task planning, navigation and tracking, recognition of objects and faces, user-interaction, among others; and (ii) a low-level to control sensors and actuators in the real world.

Bill Estranho - Hardware Description

Bill has a motion base that presents higher mobility. It is a base with 4 differential drive wheels (Mecanum Wheels). The details of each hardware are described as follows:

– **Base:** Two ESP 32 38 Pin ; Motor Controller - Ponte H L298 H-bridge Motor Driver Board 2a L298n; Mecanum Wheel; Encoder Adapter; N10-Lidar Sensor - 360°laser; ThrustMaster Controller; Bateria PowerTek 12v - End11A; One digital push button;
– **Torso**: Mini actuator Firgelli Automations; One Emergency switch;
– **Arm**: five Dynamixel-ax-12A; One ArbotiX-M; Maximum load; 1kg.
– **Head**: Intel N5095 - Mini S- 16GB+512GB; Webcam Home Office - FullHD 1080P; One Rode Videomic Pro; Watterproof Speaker - Portable Bluetooth Kimiso-112; Monitor Rear - Automotive LCD Screen 7 inches Hdmi Vga Hz-7001;

Bill Software Description

The low level is composed of a proportional control running on ESP32 boards. The communication and high level system is composed of tools developed by our team and open source applications of the Robot Operating System 2 (ROS2. The software are:

– Navigation, localization and mapping: Hector mapping.
– Face recognition: OpenCV library.
– Speech recognition: hold.
– Object recognition: hold.