Team Description Paper UFPel PinguimBots @Home LARC/CBR 2023

Kristofer Stift Kappel, Felipe Machado, Herick Dall'Agnol Nunes, Vitória de Souza Fabrício, Arthur Sandim de Oliveira, Isabele Santos Scherdien, Eduardo Porto Cavalari, Felipe Augusto Silva Martins, Nicolas Martins Claus, Raissa Nunes Coelho, Matheus Silva, Felipe de Souza Marques, and Paulo Roberto R. Ferreira Jr.

Abstract— **This Team Description Paper (TDP) introduces The Home-Environment Technological-Agent (Theta), a robot developed by the UFPel PinguimBots @Home team. The goal of the team is to inspire research in robotics, computer vision, AI, and human-robot interaction among UFPel students. Theta is built over a motorized wheelchair from Freedom Veículos Elétricos, facilitating future integration of developed technologies into potential products. Equipped with a range of sensors and processing devices, Theta can perform tasks such as mapping, navigation, self-localization, skeleton tracking, object and face detection, and Human-Robot Interaction. The team keeps continuous development and upgrade of Theta's features.**

I. INTRODUCTION

In the evolving landscape of robotics, the demand for robotic systems capable of assisting with domestic tasks has increased significantly. In response to this need, UFPel PinguimBots was established in 2019 with the goal of advancing the state-of-the-art in robotics and providing a platform for students to engage in a robotic-learning environment, with a focus on competing in LARC/CBR competitions. Service robots have emerged as a viable solution to address a diverse range of tasks within the house. The quick advancements in robotics and artificial intelligence urge for innovative solutions across various domains. One prominent area of focus is the automation of domestic tasks, with the aim of elevating the overall comfort of the home environment.

The UFPel PinguimBots@Home team, in collaboration with Freedom Veiculos Elétricos LTDA, has engaged in the research and development of The Home-Environment Technological-Agent (Theta), an autonomous service robot adapted to address domestic tasks. This innovative robot is built upon a modified wheelchair provided by Freedom, combining the domains of service robotics and autonomous wheelchairs.

To meet the challenges of the LARC/CBR competition, Theta is equipped with specialized sensors and processing devices. The YDLIDAR X4 and rotary encoders enable localization and mapping, while a Microsoft Kinect facilitates computer vision-related tasks by providing RGB and Depth information. Human-Robot Interaction (HRI) is accomplished through two interfaces: voice interaction utilizing a microphone and a speaker, and feedback expressed through facial emotions displayed on a vertically positioned monitor. The current version of Theta is depicted in Figure 1, showcasing its integrated capabilities. With these advancements, Theta seeks to fulfill the growing need for an efficient autonomous system capable of addressing diverse domestic tasks.

Fig. 1: Theta version 1.0

In this project, continuous data traversal between various applications and devices is crucial. To achieve this, Theta relies on two processing units, a computer and an ESP32. The computer is powered by Ubuntu 20.04 and ROS Noetic [1], assuming responsibility for processing tasks, such as mapping, localization, navigation, speech recognition, voice synthesis, computer vision, and rendering facial expressions. Meanwhile, an ESP32 unit plays a key role in transferring odometry data from the encoders to the computer, while also facilitating the transfer of velocity messages generated by the computer to the motor controller.

This paper is organized as follows: Section II briefly describes the team. Section III presents Theta's construc-

The work of the team was financed and supported in part by Freedom Veículos Elétricos LTDA.

Programa de Pós-Graduação em Computação (PPGC), Centro de Desenvolvimento Tecnológico (CDTec), Universidade Federal de Pelotas (UFPel), Pelotas, RS, Brazil kskappel@inf.ufpel.edu.br

tion and navigation modules. Section IV introduces the approaches to computer vision tasks. Section V elucidates the Human-Robot Interaction. Section VI talks about the simulation. Section VII draws conclusions. Lastly, Theta's hardware and software are summarized.

II. UFPel PinguimBots @Home

PinguimBots[1][2] is an interdisciplinary group from the Universidade Federal de Pelotas (UFPel) in Rio Grande do Sul, Brazil, with a dedicated focus on exploring and advancing the field of robotics. Established in 2019, the group's mission is to stimulate learning and knowledge development in robotics. PinguimBots@Home, made its debut appearance at the LARC/CBR competition in 2019, where it secured fourth place. The team returned to compete again in the following years.

The team emerged from a collaboration between UFPel and Freedom Veiculos Elétricos LTDA, a wheelchair manufacturing company. The primary objective of the team is to conduct research and develop robotic technologies designed for domestic environments. Exploiting the task-solving format of the competition, the team seeks to apply these advancements to real-world wheelchairs, expanding the potential benefits to users.

To facilitate seamless integration between the developed technologies and the wheelchairs, Theta's base frame consists of a slightly modified wheelchair from Freedom. This design choice enables the retention of the wheelchair's joystick, which serves as an interface for velocity messages. This strategic decision simplifies future integration with Freedom's products.

Currently, the team is undergoing a process of redesign of the robot, either in the physical structure of the robot, and the software and simulations.

III. THETA'S DESCRIPTION

In this section, we introduce the specifications of Theta. The primary processing power comes from a laptop computer on board the robot. The laptop. Running on the laptop is Ubuntu 20.04, along with ROS Noetic [1]. Thus, ROS offers essential features like message-passing, language independence, and modularity. These attributes are essential to the development of various interconnected modules, allowing the team to work concurrently on different aspects of the robot.

Subsection III-A gives a brief introduction about Theta's structure and base construction. Additionally, subsection III-C explains how odometry is made. Furthermore, subsection III-D explains navigation and movement.

A. Base Frame

Theta is built upon a wheelchair provided by Freedom Veículos Elétricos, a company that supports the project. Although initially designed to accommodate humans, the wheelchair's frame went through slight modifications to transform it into a robot-like structure capable of handling various domestic tasks. Theta has two pairs of wheels . The rear pair consists of motorized wheels responsible for providing odometry and being controlled by the navigation system. On the other hand, the front pair consists of swivel wheels, contributing to the robot's overall stability during operation.

As shown in Figure 1, Theta displays the integration of these components. Furthermore, the team wants to build a new torso made of steel and acrylic, aiming to improve the use of space and organization of the hardware. Nevertheless, the design remains focused on achieving optimal performance for home-environment tasks.

The joystick that was originally used as a movement interface on a conventional Freedom wheelchair remains a crucial control mechanism for Theta. This decision was made to facilitate future integration of Theta's technologies with other wheelchairs. By repositioning the joystick to a more accessible location, Theta can also be controlled by interacting directly with the joystick.

At the same time, the computer can control the wheelchair frame by publishing velocity messages unto a certain rostopic. The ESP unit then receives the message and translates it to PWM signal and sends it to the joystick controller which acts upon the wheels.

The same ESP is responsible for receiving pulses generated by both rotary encoders, translate it to odometry message and publish those messages unto the appropriate rostopic.

This dual control system allows for integrated navigation of the robot. Furthermore, the retention of the joystick interface simplifies the implementation of Theta's capabilities on potential future products, making it a practical and versatile choice for the overall design.

At the moment, this module that controls the chair is being redesigned, the code is being refactored, and document, and a new circuit is under development.

Theta draws its energy from a standard Freedom wheelchair battery, and a fail-safe button placed within easy reach acts as a safety feature, cutting off power to the motors and activating the brakes in emergencies. The standard wheelchair battery was adapted to power all onboard devices, ensuring a reliable energy supply to execute Theta's tasks efficiently.

B. Drive Control

The communication between the computer and the wheelchair is given by ROS, running on a ESP32, using PWM to control the joystick, which controls the robot. In ROS, a topic is a channel used for communication between different nodes. In this scenario, there is a ROS topic that publishes messages related to joystick commands or control signals to move the wheelchair.

ESP32 microcontroller is used to receive the messages from the ROS Topic. The ESP32 is capable of connecting to the ROS network and subscribing to the specified topic. Whenever there's a new message published on the topic, the

[1] `http://www.pinguimbots.com/`

[2] `https://github.com/pinguimbotsathome/Start_here/wiki`

ESP32 reads it and extracts the control commands from the message.

After reading the control commands from the ROS Topic, the ESP32 processes the information and converts in the analog control signals using the digital to analog converter (DAC) functionality. These analog control signals are then sent to two General Purpose Input/Output (GPIO) pins on the ESP32.

The signal from the ESP32's GPIO are passed through a buffer circuit. This buffer serves two main purposes:

- Impedance Match: It ensures that the output impedance of the ESP32 is matched with the input impedance of the downstream circuitry, optimizing signal transfer and minimizing reflections.
- Protection: It may include components like resistors, capacitors, and other protective devices to safeguard the ESP32 and the downstream circuitry from potential voltage spikes or overcurrent situations.

Both GPIOs represent axis in a cartesian plane, X to turn left and right, and Y to go forward or reverse, e.g., to move forward we need one GPIO (X) to be centered, and the other (Y) to maximum amplitude, and to turn left Y stays centered and X to maximum amplitude.

The joystick's movements, now represented as analog signals, are further processed by the circuitry responsible for controlling the motors or actuators of the chair. This circuitry interprets the analog signals and drives the motors in the appropriate direction and speed to move the chair accordingly.

In summary, this system utilizes ROS communication to transmit control commands, which are then received and processed by the ESP32. The ESP32 converts the digital commands to analog signals, which are buffered for impedance matching and protection before being used to control a joystick. The joystick's movements are then interpreted by the chair's control circuitry to move the chair in the desired direction based on user input.

C. Odometry

The integration of Hall effect sensors with the ESP32 micro-controller facilitates an efficient method for measuring the speed and direction of a rotating wheel. These measurements are accomplished through the utilization of two Hall effect sensors for each motor.

The speed detection system involves integrating two Hall effect sensors placed close to the wheel's motor shaft. Thus, a circular piece is attached to this shaft with a magnet onto it. The Hall effect sensors detect the magnetic field changes as the wheel rotates and generate pulses in response. These pulses are then transmitted to the ESP32 micro-controller through its GPIO (General Purpose Input/Output) pins.

To determine the wheel's speed, the ESP32 processes the time interval between two consecutive pulses received from one of the Hall effect sensors. This time interval is directly proportional to the wheel's rotational speed. A mathematical formula is utilized to convert the time intervals into speed measurements. The arrangement of the two Hall effect sensors at a 90° angle comes handy when detecting the direction of rotation. As the wheel turns, one sensor will produce a pulse before the other. Based on the order in which the pulses are received, the ESP32 identifies the direction of rotation.

All the data processing and direction determination occur within the ESP32 micro-controller. The final speed and direction values are then published as a ROS topic, making them accessible for other ROS nodes and applications.

D. Navigation, Localization and Path Planning

The navigation task involves moving the robot from a starting position to a designated destination point within a defined environment. To achieve this, the robot relies on a map containing the geometries of the environment and its own measurements for localization [2]. Throughout the process, the robot continuously updates its position, ensuring accurate self-localization as it traverses the path towards the destination.

The team uses the slam-gmapping node from the Gmapping package to perform mapping tasks, combining laser scan data and odometry information to generate the map, facilitating navigation and localization within the environment.

To accomplish the localization task, Theta utilizes the Adaptive Monte Carlo Localization (AMCL) algorithm. AMCL is a classical probabilistic localization system available in the ROS Navigation Stack. By employing a particle filter, AMCL estimates the position and orientation of the robot within a known 2D map, as illustrated in Figure 2. This algorithm enables Theta to effectively localize itself during tasks, ensuring accurate and reliable navigation within the given environment.

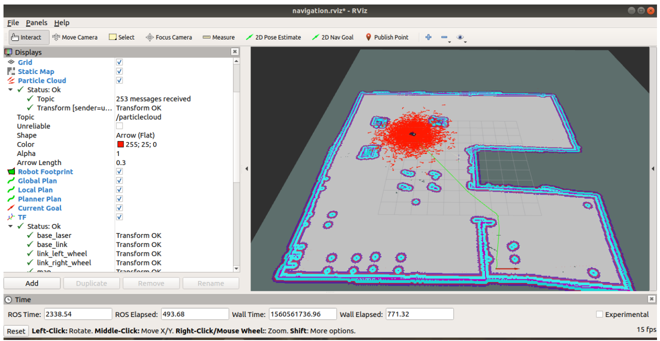


Fig. 2: Localization using Adaptive Monte Carlo Localization

Theta's path planning is done via the move-base node, which offers a configurable interface within the ROS Navigation Stack. This node is equipped with both global and local planners, components that enable the robot to perform navigation. By interacting with the move-base node, Theta has access to various functionalities within the ROS Navigation Stack, enhancing the robot's overall navigation capabilities.

RViz plays a essential role in visualizing crucial aspects of the robot's operation, such as the occupancy grid, costmaps, robot frame, and sensor readings. As a core tool within ROS, RViz also provides a graphical interface for easy interaction with the Navigation Stack, facilitating tasks like configuring initial and destination points Its user-friendly interface enhances the overall visualization experience, aiding in data interpretation to optimize robot performance. An illustrative example of RViz running AMCL can be observed in Figure 2.

IV. Robot Vision

Computer Vision plays an important role in the functionality of a domestic robot as it enhances the robot's perception of its surroundings and aids in informed decision-making.

The subsequent subsections will showcase Theta's computer vision capabilities, which enable it to better interact with and understand its environment.

A. Object detection

With the ability to detect and identify objects in its environment, Theta can effectively interact with its surroundings, contributing to its hability in domestic tasks.

In the 2020 LARC/CBR edition, the team employed a ResNet 18 Neural Network implemented with the PyTorch framework, for object detection tasks. The team built upon and enhanced the approach presented in [3] to approach the specific requirements of the competition.

However, during the actual competition, the performance encountered certain challenges, including errors such as low precision rates and shifted bounding boxes. These outcomes led the team to explore alternative approaches to improve the object detection and recognition capabilities of Theta.

Recognizing the need for a more effective object detection, the team transitioned to an R-CNN model [4]. The team adapted and built upon the content available in [5], which utilizes the Keras library on the TensorFlow platform, which further was adapted to learn multiple classes during training. This new approach seeks to address the challenges faced during the previous competition and aims to enhance the precision and accuracy of Theta's object detection. By exploring and adapting different models and techniques, the team continues to advance Theta's capabilities, trying to achieve superior performance and results in future competitions and real-world scenarios.

The model consists of an R-CNN model pre-trained over the imagenet dataset. The input is a RGB image with a 224x224 dimension.

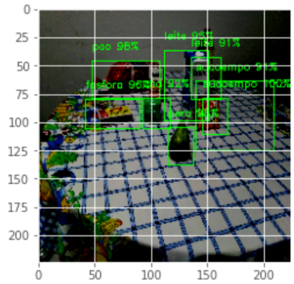
The training dataset is made up from images gathered from the past editions of the competitions. Thus, the images contained objects from fifteen different classes (Figure 3).

B. Face Detection

Face detection is a computer vision topic that involves identifying and locating human faces in digital images or video frames. It plays an important role in numerous applications, such as detecting whether a potential operator is nearby or recognizing particular users based on their facial expressions.

Five libraries for face detection were evaluated: DLIB, DLIB+CNN, DLIB+HOG, Haar Cascade, and Mediapipe.

(a) Image with different objects used for training.



(b) Objects after classification.

Fig. 3: Images before and after classification.

Among them, Mediapipe stood out as a suitable choice. Therefore, Mediapipe was selected as the library to be implemented in the detection and recognition process.

This process will be conducted in three stages:

- Face Detection and Data Storage: We will utilize MediaPipe to analyze input images or video frames and identify faces. Subsequently, the collected data will be organized and stored in a dataset.
- Training: During this crucial second phase, we will extract all user data from the dataset and proceed to train the OpenCV Recognizer. The outcome of this training will be a yml file.
- Face Recognition: In this final stage, our camera will capture a new face. The trained recognizer will then come into play, making a prediction based on the captured data. The recognizer will return the person's id and indicate the confidence level of the match if the face has been previously captured and trained.

C. Follow Me

In a domestic environment, a typical @home task for a service robot is following a person. To achieve this, Theta utilizes a Kinect sensor, which offers both an RGB camera and a depth sensor.

To successfully follow a person, Theta relies on the integration of OpenNI and Nite with ROS. OpenNI acts as a middleware, providing access to the functionalities of the Kinect Sensor. Meanwhile, Nite is accessed through the openni_tracker package on ROS, which publishes a set of transforms representing the joints of the skeleton on the /tf topic. These transforms contain information about the user's current position, including distance and deviation from the center. With this data, Theta can continuously approach the calibrated skeleton while maintaining a predefined safe distance. By effectively utilizing these tools, Theta can autonomously track and follow a person, ensuring a safe and efficient interaction in a domestic environment.

V. Human-Robot Interaction

Human-Robot Interaction (HRI) is crucial for domestic robots to ensure a pleasant user experience. Effective communication through visual and audible feedback is vital to build trust and avoid frustration.

A. Facial Expressions

Theta employs a set of animated facial expressions to provide visual feedback, indicating its current status and emotions during interactions with the user. These expressions include emotions like happiness after successfully completing a task, sadness when unable to understand a command, or a thoughtful expression when processing information. Additionally, visual feedback such as loading screens during startup or sleep screens when inactive further enhances the interaction.

By following an Affective Loop, Theta dynamically adapts its facial expressions to match the appropriate situation during user interactions [6]. This approach keeps the user engaged and encouraged to interact more with the robot, sustaining a positive and immersive experience. The continuous change in facial expressions helps create a more natural and relatable interaction, making Theta a friendly and approachable companion in the domestic environment.

Fig. 4: Some of the facial expressions that theta can reproduce

Theta's face module utilizes a ROS topic for communication with other applications. By subscribing to this topic, Theta can receive messages indicating when to change facial expressions or the current status. This flexibility enables Theta to provide appropriate visual feedback regardless of the task or interaction at hand.

To address diverse human-robot communication needs, the team continuously adds new faces to the available expressions.

B. Speech Recognition and Synthesis

Speech recognition is the ability of a machine to understand and translate spoken words into text. It is a part of Natural Language Processing (NLP) and is used in various applications and systems to transcribe spoken words into written form.

In this project, all the tools used are libraries for the Python language. It is used Speech-To-Text and Text-To-Speech tools with Hot-word Detection to make Theta understand what is spoken and perform an action. The following items explain the tools chosen for each one of them.

1) Speech-To-Text: Speech-To-Text or Automatic Speech Recognition(ASR) is a technology that converts spoken language into written text. Several options were evaluated for STT, including Nvidia NeMo, Whisper from OpenAI, Vosk and Google Speech.

The testing phase focused on evaluating the accuracy, robustness, and overall effectiveness of the system in recognizing and transcribing spoken language. In the initial tests, Vosk had a bad performance compared to the others and was discarded. As for Google Speech, though it is a widely recognized tool, it does not meet the required criteria for this project.

Nvidia NeMo has an extensive toolkit with multiple options of models for STT, but none of the models is as accurate as Whisper‘s. Also, NeMo's model takes longer to load and recognize speech. Therefore, Whisper was chosen as the tool for STT.

2) Text-To-Speech: Text-To-Speech or Speech Synthesis converts written text into an audio output with human-like or synthetic voices. Many online libraries also have an option for STT, therefore the Nvidia NeMo was also evaluated for TTS. Other tools tested include the API gTTS(Google Text-To-Speech) and the library TTS by Coqui. Nvidia NeMo stands out among the tools due to the possibilities given for speech synthesis with its Cascaded TTS and End-to-End TTS. It can also change the frequency of the output audio and its speed. For this reason, Nvidia NeMo was chosen for TTS.

3) Hot-word Detection: Hot-word Detection, also known as WakeWord detection, is a technology used in voice-activated systems and devices to recognize specific words or phrases that act as a trigger to activate the system. The primary purpose of hotword detection in this project is to initiate the speech recognition system when it hears a predefined word.

Several Python libraries were tested, including Porcupine from Picovoice and PocketSphinx from CMUSphinx. Porcupine stands out as the most accurate during the testing phase, performing well in noisy environments. Therefore, it was chosen as the Hotword Detection tool to be used in Theta.

VI. Simulation

The team utilizes 3D modeling to create digital representations of objects, mechanical parts, and prototypes. In this manner, it is possible to join the parts the team produces and simulate the movements of objects inside simulation software. Being very effective for building robots and automated parts.

Futhermore, the team uses Onshape CAD[3], a computer-aided design software. Through the software, it was possible to recreate the full-size structure of Theta. Therefore, several parts are created separately and put together to assemble the robot. With the help of the onshape-to-robot [4] tool, which allows us to import robots designed in Onshape CAD and transform them into descriptions in URDF or SDF format, the team can use them to simulate physical properties of the robot.

Figure 5 shows the assembled Theta, after going through the onshape-to-robot tool to generate a URDF file, the file responsible for unifying the parts and being ready to use a the simulation software.

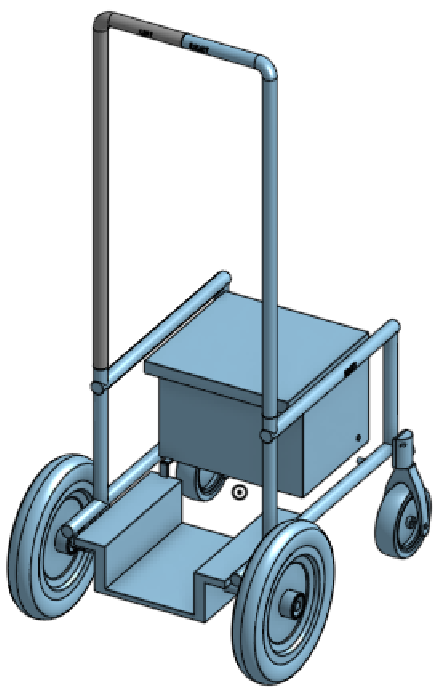


Fig. 5: Theta's 3D modelled exterior structure

VII. CONCLUSIONS

This technical description paper provides an in-depth overview of The Home-Environment Technological-Agent (Theta), a service robot developed by the UFPel PinguimBots @Home team in collaboration with Freedom. Theta is built over a wheelchair base and equipped with various sensors to handle domestic tasks and participate in LARC/CBR @Home challenges.

Future improvements for Theta include redesigning the robot frame to address current flaws, adding a manipulator and additional sensors, and enhancing Human-Robot Interaction (HRI). The team aims to extend their research to autonomous wheelchairs that can navigate independently and provide assistance in domestic environments. With ongoing development and research.

[3]https://www.onshape.com/en/

[4]https://onshape-to-robot.readthedocs.io/en/latest/

REFERENCES

[1] “Ros documentation,” http://wiki.ros.org/, 2018, [Accessed: 10-06-2019].

[2] M. Quigley, B. Gerkey, and W. D. Smart, *Programming Robots with ROS: a practical introduction to the Robot Operating System.* ” O’Reilly Media, Inc.”, 2015.

[3] “Jeremy fix’s deep learning lectures,” https://teaching.pages.centralesupelec.fr/deeplearning-lectures-build/index.html, 2020, [Accessed: 24-08-2021].

[4] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[5] “R-cnn object detection with keras, tensorflow, and deep learning,” https://www.pyimagesearch.com/2020/07/13/r-cnn-object-detection-with-keras-tensorflow-and-deep-learning/, 2020, [Accessed: 24-08-2021].

[6] K. Höök, “Affective loop experiences–what are they?” in *International Conference on Persuasive Technology*. Springer, 2008, pp. 1–12.

VIII. SOFTWARE AND HARDWARE DESCRIPTION

In this section, we summarize Theta’s hardware and software features.

A. Hardware

- The base consists on a modified wheelchair manufactured by Freedom.
- A pair of rotary encoders made from hall effect sensors.
- A Kinect camera .
- A YDLIDAR.
- A ESP 32 which communicates the rotary encoders to the computer and the latter to the joystick controller.
- A monitor, a microphone and a speaker to perform HRI.
- A notebook with an Intel Core i5 processor and 8 GB of RAM .
- A Failsafe Button that cuts the power off from the motors and activate the breaks.

B. Software

- Ubuntu 20.04 running ROS noetic Noemia.
- OpenCV and MediaPipe for image recognition tasks.
- ROS Navigation Stack and gmapping.
- Whisper and Nemo for Speech Recognition tasks.
- OpenNI and Nite to perform skeleton tracking.