CATIE Robotics @Home 2019 Team Description Paper

Rémi FABRE, Boris ALBAR, Clément DUSSIEUX, Ludwig JOFFROY, Zhe LI, Clément PINET, Jennifer SIMEON, and Sébastien LOTY

Centre Aquitain des Technologies de l'Information et Electroniques (CATIE)
1 Avenue du Dr Albert Schweitzer, 33400 Talence, France
r.fabre@catie.fr
http://robotics.catie.fr/

Abstract. This paper provides an overview of the CATIE Robotics Team's activities for participating at the RoboCup@Home 2019 in Syndey. During the first year, the main objective is to build a solid base for years to come of service robotics developments. TIAGo, an off-the-shelf robotic platform was acquired. Safe and autonomous navigation is achieved by integrating ROS compatible components. Snips, an offline, free tool is used for both speech recognition and Natural Language Understanding. Siamese neural networks are used for face recognition and person tracking. Promising results are obtained. Our robot is able to navigate safely and avoid common obstacles. It is also able to answer limited questions. Offline we have reproduced state of the art performance for face detection and recognition. We also tested state of the art libraries for body pose estimation and object segmentation. Integration of these technologies is an ongoing task on TIAGo. An approach towards object manipulation is discussed.

1 Introduction

The CATIE Robotics Team formed at the beginning of 2018 and is part of CATIE, a digital technology transfer center at the crossroads between research and industry. CATIE is a non-profit organization supported by the Nouvelle-Aquitaine region in France with the mission to support companies for adopting and integrating digital technologies in their technological and economic development. CATIE is composed of three departments: human factors, electronics and artificial intelligence.

By creating a RoboCup@Home team, our ambition is to be part of a community of experts, nurture our robotic knowledge and share it to foster progress towards a tangible goal. The competition offers an objective benchmark to measure progress and is a clear motivational tool to unite efforts locally.

The main objective in our first year is to create a solid technical foundation that will facilitate future work and collaborations. To do so, we believe that the most efficient path is to focus first on integrating proven technologies and achieving robust behaviours. The Robot Operating System (ROS) was a natural

middleware choice and TIAGo, an off-the-shelf service robot, was acquired in August 2018.

Promising results have already been achieved[1] and strong enhancements are expected in the months to come as the integration continues on the TIAGo.

In this paper, the developments surrounding the navigation, perception, communication and object manipulation capabilities of the robot are presented. For each section, an overview of the approach, the results and the future work is given.

2 Navigation and SLAM capabilities

2.1 General approach

Reaching a robust, reliable navigation is our main priority for this year. A comparison between several SLAM approaches was made and two were selected for further scrutiny. The papers behind Gmapping [1] and Cartographer [2] have been studied and their implementations have been tested on a TurtleBot3 platform[2]. The same tests were run on the TIAGo once it was available. This process has led to a good understanding of the numerous parameters of the algorithms and the impact of the quality of the sensors. In particular, good results have been achieved with Cartographer with both platforms, in localization in a known environment, map creation (see Fig.1) and full SLAM scenarios. We presented our approach at the invitation of LaBRI where we discussed the merits of the exercise [3][4].

At the time of the demonstration shown in the qualification video, the following were implemented: AMCL for the localization, ROS move_base for the navigation and a static map created with Gmapping. A near-term goal is to continue the integration of the previous work into TIAGo's navigation pipeline to obtain full SLAM capabilities during the autonomous navigation and not only during the cartography process.

Given the strong embedded-systems expertise of CATIE, a will was born to obtain a stand-alone, power efficient, embedded solution for the robot. While this dimension was not a priority in the first year, a new SLAM-capable, ROS-agnostic, hardware accelerated platform was tested[5] with good results.

2.2 Obstacle avoidance

While the main sensor for performing SLAM is the 2D LIDAR, it is not enough to ensure safety during the navigation. Tables are a good example of obstacles that are poorly detected. We enhanced the obstacle avoidance capabilities of our

[1] `https://www.youtube.com/watch?v=9YdRQpUKalw`
[2] `http://emanual.robotis.com/docs/en/platform/turtlebot3/overview/`
[3] `http://aspic.labri.fr/`
[4] (French) `http://aspic.labri.fr/slides/cartographer.pdf`
[5] `https://www.youtube.com/watch?v=vJo_XOrTmGs`

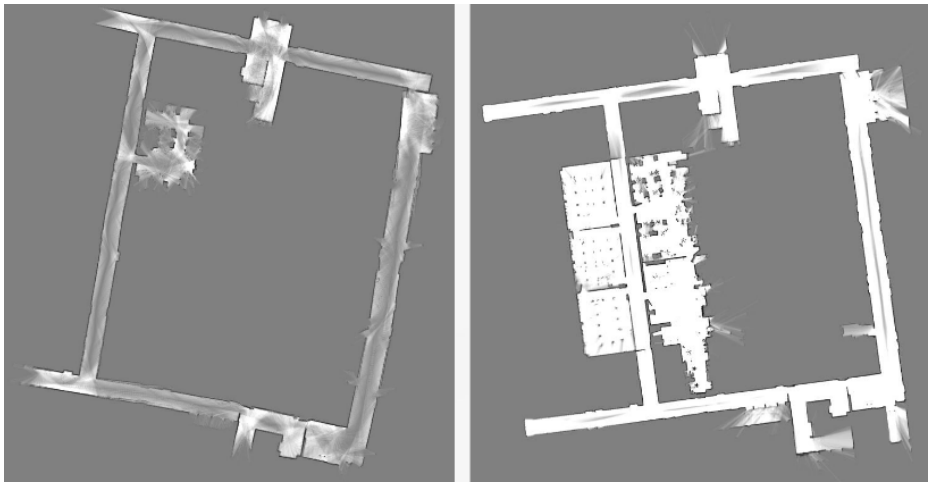

Fig. 1. (left) Map obtained on the TurtleBot3. (right) Map obtained on the TIAGo. Both using Cartographer in the same building, the longest corridor is 40m long

robot by using the depth information of the RGB-D camera: the points above the LIDAR and below the height of the robot are projected into an horizontal plane and used as obstacles in the local and global cost maps. Also, since both sensors are on the front of the robot, no backward motion is allowed. Instead, most motions will start by a rotation. This is possible since the robot's base is round and the arm can be tucked in during navigation.

The current setup has been tested around glass doors, tables, walking people and partially transparent barriers without a single collision.

3 Perception capabilities

Vision is handled by several different State of the Art neural networks detailed below. For each neural network, we tried to find the best compromise between accuracy and performance.

3.1 Siamese networks

Siamese networks are a type of neural networks that learn a similarity measure between objects[3]. Every object is mapped by the network to a vector of fixed size (typically 64 or 128 in our case). Then we can compute the similarity between two objects as the distance between the two vectors (euclidean or cosine distance). Siamese networks are used in our context for different purposes including face recognition and people tracking.

For vision problems, we used convolutional neural networks as they are the state of the art in this domain. However, inference in large convolutional neural networks are impractical as they require significant computing power. We used a modified version of ResNet-18 which is a residual network with 18 convolutional layers. ResNet-18 is one of the smallest neural networks in its class and is a good compromise between performance and speed (it performs one iteration in 0.008sec on a Jetson TX2). Moreover, all these networks have been ported to TensorRT which allows an even greater inference speed.

We trained these networks from scratch on public databases. To accelerate the training and make the result more robust, we make use of several tricks. The network is first trained on the classification problem (i.e. recognizing one person between all possible identities) and then the trained network is refined to learn an embedding by changing the last layer and the loss function. This allows us to accelerate considerably the training of the network. Moreover several data augmentation techniques are used to help generalization of the model and make the neural network more robust to noise by adding Gaussian noise to the input image and penalizing the loss function with the KL-divergence (or L_2 distance) between the noisy and the normal image.

3.2 Face detection and recognition

Face detection and recognition is done by a combination of two neural networks. The first one is a network that is used out of the box and that performs the face detection using a state of the art multi-task cascaded convolutional network (MTCNN). Once the face bounding boxes are extracted as well as face features (eyes, nose, ...), the image is cropped and normalized. The faces are then passed through a siamese network. This network has been trained on more than 8000 different identities using the publicly available VGG 2 dataset as described above. The distance between the faces and the previously detected people are computed and the closest person in term of distance is returned if the distance is below a certain threshold.

3.3 Person tracking and follow-me behaviour

The follow-me behaviour displayed in the qualification video uses a simple approach based on the cloud points of the RGB-D camera. Consider an immobile 3D box in the robot's base frame (X axis is towards the front of the robot, Y axis is towards its left), positioned so that the human to follow should be in the center of it at all times. The centroid of the cloud points inside the box is calculated and two PID controllers are implemented. The first one controls the linear speed of the robot, the error being the difference along the X axis between the centroid and the center of the box. The second one controls the rotational speed of the robot, the error being the difference along the Y axis between the centroid and the center of the box.

While this approach offers some flexibility (position and size of the box, PID coefficients), we are in the progress of coupling it with a siamese network in order to improve its robustness[6]. The idea is similar to the face recognition problem. First, people are extracted from the image using an object detection neural network, in our case YOLO. Then we use a siamese neural network to measure the distance between every person in the frame and the person we want to track. We currently use a moving average of the previously detected vectors as a reference for computing the distances. This allows the following of changes in the person orientation or size of the person during long tracking sessions and thus largely improve the stability of the algorithm. This approach could be used to track any type of object by generalizing the network to learn features independent of the object type.

We have the ambition to improve YOLO and to use it in other use cases. In the near future, we plan to train it to detect people and recognize specific objects as well. One drawback is that YOLO outputs large bounding boxes. For cases of object detection, the large bounding box does not pose a problem, but for the task of object-grasping a more precise image segmentation would be necessary. For the object-grasping task we need the object's exact contour (masks) to decide the best way to grasp the object. A current project is to adapt YOLO to perform object segmentation to yield exact masks of the objects instead of a bounding boxe without compromising on YOLO's inference speed. Another problem with YOLO is its recall as it tends to classify one object on one frame and forgets to find it on a following frame. YOLO's precision has also been found to be unreliable. We hope to improve this by training the neural network on a video dataset and penalize the loss of the network accordingly to improve the model.

3.4 Pose Recognition

The pose recognition is done with the OpenPose library ([7]) that computes face, body and hands 2D keypoint detection in real-time if equipped with a recent graphical card. For now, we only use the Body-Foot Estimation (default skeleton) in order to maintain a frequency that is adequately high (the algorithm runs at around 7 FPS on a Nvidia Jetson TX2). In the future, we may use hands estimations also in order to detect gestures; for example, if a person is pointing towards an object. Once the skeleton is computed, the pose can be determined by simple rules, such as: "If the shoulders are approximately at the same height than the hips and the neck is significantly higher, then the person is sitting". We wrote this code on top of OpenPose to categorize the person's pose (sitting, standing, raising arm, etc.). While we have not integrated this on the complete robot behaviour yet, we have achieved results that seem robust enough to answer most pose-related SPR questions. Also, handling the waving gesture recognition with the static pose approach works with acceptable performances.

[6] See the "person tracking" section of the qualification video

[7] `https://github.com/CMU-Perceptual-Computing-Lab/openpose`

4 Communication capabilities

4.1 Text-To-Speech, Natural Language Understanding and speech synthesis

For the speech-to-text and Natural Language Understanding (NLU), we are using Snips [8]. Snips is an off-the-shelf solution, free for non-commercial use software. While its accuracy is less than other well-known cloud-based solutions, both the speech recognition and the NLU engine run offline with low latency, which is vital for the RoboCup competition.

Snips uses very few resources that it can even be embedded on a Raspberry Pi. One of the drawbacks is that the speech-to-text does not appear to be generalized: working well with phrases that resemble what has been learned, but failing at times when working with a new sentence pattern.

To work properly, Snips needs to collect training samples with each use case. To acquire the data, we used the Robocup Speech and Person Recognition (SPR) and General Purpose Service Robot (GPSR) questions generator[9]. Consider the spoken question : "Where can I find the apple?". Snips will translate it to text and then analyze it. It will match the trained *user intention* called "askObject-Location" and detect "apple" as parameter, then it will post the result in an MQTT topic. Every step of the Snips process is linked to a specific MQTT topic, this is a very convenient feature to keep control over the stack[10]. Custom code will then take over and call the appropriate functions.

With this tool the robot is able to translate and categorize any SPR or GPSR question. Currently, only SPR questions related to object category/location and predefined ones are answered by the robot. The knowledge base is handled through XML files that are similar to the ones used in the SPR and GPSR generators. We plan to integrate the person and object recognition based questions in the months to come.

For text-to-speech, we are using Tiago's default text-to-speech module: Acapela[11].

5 Object manipulation capabilities

In order to obtain a working demonstration on the robot rapidly, the grasping tasks have been omitted for now. That being said, the capability is of high interest for the team. Several projects have been started on the subject:

- A proof of concept of an universal gripper has been made. An embedded prototype is being built on the same principle. If the results are satisfactory, the prototype will be integrated on the robot as an alternative gripper[12][13].

[8] `https://snips.ai`
[9] `https://github.com/kyordhel/GPSRCmdGen`
[10] `https://snips.gitbook.io/documentation/ressources/hermes-protocol`
[11] `http://www.acapela-group.com/`
[12] `https://www.youtube.com/watch?v=mMfReo0AEbM`
[13] `https://www.youtube.com/watch?v=u9xp1qkpyr8`

- Two projects have been started in collaboration with the Bordeaux INP robotics specialization students, it will be their main project for the fall semester 2018. The objective is to be able to grasp known objects on a table. The first project focuses on the path planning of the arm, torso and head with MoveIt! in order to grasp a known item without collision, using an Octomap (3D occupancy grid) created with the RGB-D camera[14]. The second project focuses on the perception part of the problem, training YOLO to recognize the set of items and segment the 3D occupancy grid accordingly.
- We are in the process of acquiring PAL Robotics' implementation of an hierarchical quadratic solver based on the stack of tasks that handles the 7DoF arm, the torso prismatic joint and the 2 DoF head[4].

6 RoboCup experience and community outreach

While the team is very young, its team leader has continuously been involved in RoboCup activities since 2015. He is a former member of team Rhoban and participated in three RoboCup Soccer Kid-Size competitions: in 2015 the team achieved third place and in 2016 and 2017 the team achieved first place[5][6][7][8][9]. An open source implementation of an advanced motor control firmware was presented in RoboCup's 2016 Symposium[10].

We have been invested in the organization of the first two editions of the French RoboCup Junior. We participated in Montesilvano's 2018 RoboCup@Home Education challenge and one of us was member of the jury in Montreal's edition of the same workshop. We were present at the 2018 RoboCup competition to connect with the @Home community and help AMC, a sister team in its first RoboCup participation in the SSL league. Also, we have been active on the rule book discussion[15].

We were exhibitors at Cap Sciences' Village des Sciences, that gathered more than 3000 people over the weekend around robotics and the RoboCup competition[16]. We're planning the organization of the RoboCup@Home Education challenge in Bordeaux in early 2019.

7 Conclusions

In this paper, we have given an overview of the approaches used by the team CATIE Robotics for the RoboCup@Home competition. The objective in this first year is to obtain robust and reliable behaviours on the robot, especially in the fields of navigation and vocal communication that we believe are crucial components to enable solid service robotic developments. State of the art neural

[14] http://wiki.ros.org/Robots/TIAGo/Tutorials/MoveIt/Planning_Octomap

[15] https://github.com/RoboCupAtHome/RuleBook/issues?utf8=%E2%9C%93&q=author%3ARemiFabre+

[16] http://www.cap-sciences.net/au-programme/evenement/village-des-sciences-2018

network architectures have been successfully tested for face recognition, person tracking, pose recognition and object segmentation. An approach towards object manipulation has been identified and will be worked on this year. Encouraging results have already been achieved with our chosen strategy. While advancements have been made, more work is needed to accomplish our long-term goals. Clear, short-termed enhancements are to continue integration of features we have tested. We plan to participate in several competitions in the coming year to gain experience. Finally, a CIFRE PHD thesis will start this year on embedded hardware acceleration applied to the training and inference of neural networks, we hope that this work will enhance the capabilities of our robot in the future.

References

1. Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. Improving grid based slam with rao blackwellized particle filters by adaptive proposals and selective resampling. *IEEE International Conference on Robotics and Automation (ICRA)*, 2005.
2. Wolfgang Hess and al. Real time loop closure in 2d lidar slam. *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.
3. Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition.
4. Mansard, O Stasse, P Evrard, and A Kheddar. A versatile generalized inverted kinematics implementation for collaborative working humanoid robot: the stack of tasks. icar 2009.
5. Julien Allali, Louis Deguillaume, Rémi Fabre, Loïc Gondry, Ludovic Hofer, Olivier Ly, Steve N'Guyen, Grégoire Passault, Antoine Pirrone, and Quentin Rouxel. Rhoban football club: Robocup humanoid kid-size 2017 champion team paper. 2016.
6. Julien Allali, Rémi Fabre, Loïc Gondry, Ludovic Hofer, Olivier Ly, Steve N'Guyen, Grégoire Passault, Antoine Pirrone, and Quentin Rouxel. Rhoban football club: Robocup humanoid kid-size 2017 champion team paper. 2017.
7. Rémi Fabre, Hugo Gimbert, Loïc Gondry, Ludovic Hofer, Olivier Ly, Steve N'Guyen, Grégoire Passault, and Quentin Rouxel. Rhoban football club - team description paper humanoid kidsize league, robocup 2016 leipzig. 2016.
8. Julien Allali, Rémi Fabre, Hugo Gimbert, Loïc Gondry, Ludovic Hofer, Olivier Ly, Steve N'Guyen, Grégoire Passault, Antoine Pirrone, and Quentin Rouxel. Rhoban football club - team description paper humanoid kid-size league, robocup 2017 nagoya. 2017.
9. Julien Allali, Rémi Fabre, Hugo Gimbert, Loïc Gondry, Ludovic Hofer, Olivier Ly, Steve N'Guyen, Grégoire Passault, Antoine Pirrone, and Quentin Rouxel. Rhoban football club - team description paper humanoid kid-size league, robocup 2018 montreal. 2018.
10. Rémi Fabre, Quentin Rouxel, Grégoire Passault, Steve N'Guyen, and Olivier Ly. Dynaban, an open-source alternative firmware for dynamixel servo-motors. *RoboCup 2016: Robot World Cup XX*, 2016.
11. Si Li and Wu Li. A unified multilanguage recognition system. *The Unique Journal in Advanced Robotics*, 22(01):42–69, 2098.

Robot TIAGo Hardware Description

Robot TIAGo has been selected and is being customized for the @home competition purpose. Specifications are as follows:

- Base: differential drive base, 1m/s max speed.
- Torso: lifting torse (35cm lift stroke)
- One arm with a gripper (7 DoF). Maximum load: 2kg.
- Head: 2DoF (pan and tilt)
- Robot dimensions: height: 1.10m - 1.45m, base footprint: 54cm diameter
- Robot weight: 72kg.

Our robot incorporates the following sensors:

- RGB-D camera
- 2D LIDAR
- Stereo microphone
- Speaker
- Sonars
- IMU
- Motors current feedback

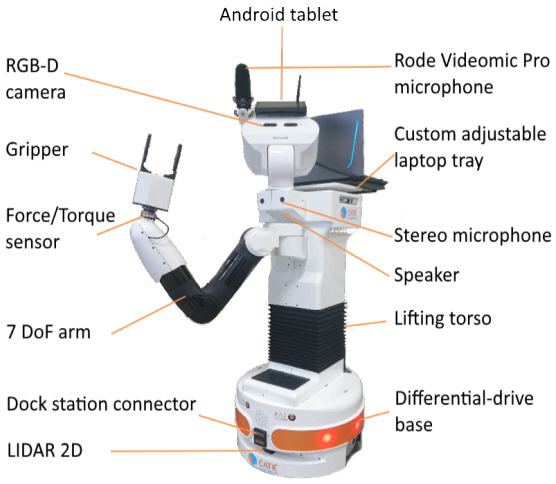

Fig. 2. Robot TIAGo

Robot's Software Description

For our robot we are using the following software:

- OS: Ubuntu 16.04
- Middleware: ROS Kinetic
- Simulation: Gazebo
  http://gazebosim.org/
- Visualisation: RViz
  http://wiki.ros.org/rviz
- Localization: AMCL
  http://wiki.ros.org/amcl
- SLAM: Cartographer and GMapping
  https://github.com/googlecartographer/cartographer
  http://wiki.ros.org/gmapping
- Navigation: move_base
  http://wiki.ros.org/move_base
- Arms control: moveIt! and play_motion
  http://moveit.ros.org/
  http://wiki.ros.org/play_motion

- Face recognition: custom siamese neural network
- Object recognition: YOLO
  https://pjreddie.com/darknet/yolo/
- Pose detection: Open Pose
  https://github.com/CMU-Perceptual-Computing-Lab/openpose
- Speech recognition: Snips
  https://snips.ai/
- Speech generation: Acapela
  http://www.acapela-group.com/
- Task executor: SMACH
  http://wiki.ros.org/smach

External Devices

Our robot relies on the following external hardware:

- Rode Videomic Pro external microphone
- External laptop.
- Android tablet