

RoboIME's walking Humanoid

Luis D. P. Farias¹, Fábio Suim Chagas², and Ana L. B. Silva³ José L. O. Schramm⁴ Mateus S. Carvalho⁵ Gabriel Lima⁶ Marcos Paulo⁷ Matheus Bozza⁸ Onias C. B. Silveira⁹ Paulo F. F. Rosa¹⁰

Instituto Militar de Engenharia, rpauloime@gmail.com
<http://www.roboime.com.br>

Abstract. This paper describes the robot of Humanoid from RoboIME Team and its purpose of demonstration at RoboCup. It contains the platform configuration and walking algorithm.

Keywords: Humanoid · Humanoid Robot Racing · IA learning

1 Introduction

RoboIME is the team of the robotics laboratory of IME (Military Engineering Institute) that participated in several editions of LARC (Latin American Robotics Competition) and RoboCUP, where it achieved good results in the SSL (Small Size League) category as first place in LARC 2017 and second place in division B of RoboCup 2018-2019. Thus, continuing the progress of the laboratory, the team started to develop its first humanoid robot in the last year where it managed to participate in Larc through the IEEE Humanoid Robot Racing category (IEEE HRR) and get fourth place. For the demonstration at this 2020 RoboCup in Bordeaux, the team intends to demonstrate the technical challenge of the IEEE HRR where the robot has to complete the track avoiding characteristic obstacles.

2 The Robot

2.1 Electronic Project

The electronic design of the robot consists of a central processing unit, sensors, motors and a battery.

- The Central Process Unit consists Raspberry Pi 3.
- The sensors used was a force sensitive resistor (FSR), MPU9250 as accelerometer and gyroscope, TF Mini's as distance sensor and for vision we plan to use a camera module for Raspberry Pi.
- For the power supply, we used the battery TURNIGY 5000mAh 3S 20C Lipo Pack with XT-60.
- The motors used are the DYNAMIXEL AX-12A.

The firmware consists of a python library created to control motors and sensors.

2.2 Mechanical Project

The mechanical design was developed looking for similar proportions of a human being but limited by fixed components such as the battery, the motors and mainly by the rules of competition.

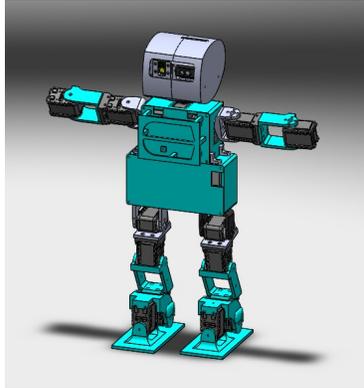


Fig. 1. Robots CAD.

3 Simulator

To elaborate the joints path planning we developed a simulator by using Matlab's Simulink module software, In order to test it quickly and safely.

For this we used the Simscape Multibody module that allows importing the robot from Solid Works and an additional library, Simscape Multibody Contact Forces Library, to simulate the contact between the robot's feet and the ground. The simulator code is divided into 3 main parts:

- The robot assembly is responsible for the physical part of the robot.
- The movement part describes the movement calculated by the learning program that generates the robot's trajectory.
- And finally the part of the analysis of stability that will send information to this same program.

While the robot is being modeled, a simplistic model was used to test the simulator and the learning code, but once it is finished it will be replaced by the actual model that was shown. A view of the simulator is shown in figure 4.

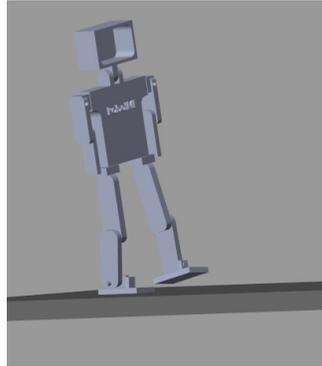


Fig. 2. Explorer view of the Mechanic Simulator

4 Motion planning

For the first model of walking, we adopted the model of statically stable movement, in this model the CoM(Center of Mass) always remains within the SP¹ (Support Polygon) during the entire motion. This implies that if the movement is stopped, the robot will remain in a stable position. This kind of stable movement is only valid for low walking velocities, which impose also low angular velocities in the joints[3].

For this, the full-body posture goal method was used. In this method, a set of angles of the robot's joints are defined, which determine its configuration so that each of these positions is stable and its sequence forms a walk. The algorithm consists in going from one configuration to another in a random way and verifies if this way was satisfactory, so far we only check if the robot falls or not. If the robot falls, another way it is chosen for him to go from one configuration to another. This repeats itself until it reaches the final configuration, thus completing its walk.

Because this algorithm requires several tests, the simulator becomes essential.

5 Computational Vision

Computer Vision is being developed with the help of the OpenCV library. It allows us to identify the edges present in the image using the FAST (Features from Accelerated Segment Test) algorithm. Analyzing characteristics such as size and orientation of these borders, highlighted in figure 3, it is possible to obtain information such as the position of the field boundary and possible obstacles.

¹ Support Polygon is the convex polygon formed by the floor support points during robot walking.

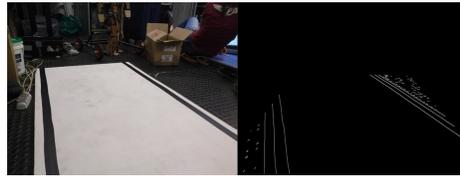


Fig. 3. Real image vs Computer Vision image.

6 The Demonstration

The technical challenge demands the robot to walk through the arena avoiding obstacles of different heights. The obstacles are placed in random places over the arena and the robot must try to complete the arena by evading them without let them fall.

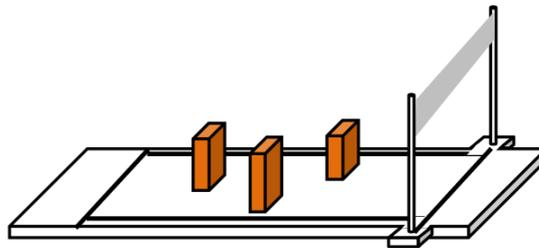


Fig. 4. Field of Humanoid Robot Racing.[2]

7 Conclusions

This article resumes the work of RoboIME's Humanoid Team to participate in the showcasing at RoboCup 2020. We developed our own simulator, walking algorithm, and all robot design.

References

1. M.H.P. Dekker,ZERO-MOMENT POINT METHOD FOR STABLE BIPED WALKING. NEindhoven, July (2009).
2. IEEE Latin American Robotics Student Competition, HUMANOID ROBOT RACING RULES, – January (2016).
3. ZERO-MOMENT POINT METHOD FOR STABLE BIPED WALKING, M.H.P. Dekker, NEindhoven, July, 2009.