

Rhoban Football Club – Team Description Paper

Humanoid KidSize League, Robocup 2016 Leipzig

R. Fabre, H. Gimbert, L. Gondry, L. Hofer,
O. Ly, S. N'Guyen, G. Passault, Q. Rouxel
`remifabre1800@gmail.com`, `gimbert@labri.fr`, `loic.gondry@free.fr`, `lhofer@labri.fr`,
`ly@labri.fr`, `steve.nguyen@labri.fr`, `g.passault@gmail.com`, `quentin.rouxel@labri.fr`

CNRS, LaBRI, University of Bordeaux 1
33405 Talence, FRANCE

Abstract. This paper gives a short overview of the design of the Kid-size humanoid robots Sigmaban and Grosban of the French *Rhoban Football Club* Robocup team. These robots are built to play soccer in an autonomous way. It describes the main hardware and software components in their current state, major upgrades and research tracks for the upcoming Robocup 2016 competition.

1 Introduction and Last Participations

Rhoban Football Club ¹ [9] is an on-going robotic project whose team members are researchers and PhD students at University of Bordeaux (France) and CNRS.

The interest of the team is mainly autonomous legged robots and their locomotion. Our two leading projects are a small and low cost open source quadruped robot ² and kid to mid size humanoid robots with the RoboCup competition as major ambition (Sigmaban and Grosban platform). In this context, several prototypes have been built and tested [4,5,3,1] with a special emphasis on pragmatic and operational solutions.

The very challenging problem of robots playing autonomous soccer in complex and semi-unconstrained environment has driven the team to propose new mechanical designs – spine-oriented robot have been tested, low-cost foot pressure sensors are experimented – and software methods – new custom servo-motors firmware, learning algorithms applied to odometry.

Our participation to Robocup 2016, up to the qualification procedure, would be the fifth one:

- 2011 (Istanbul): Very first participation of the team at RoboCup competition under the name *SigmaBan Football Club*.
- 2013 (Eindhoven): Second participation under current name *Rhoban Football Club*. For the first time, the team was able to submit three robust humanoid robots without major hardware problem.

¹ The page of the team is accessible at:

<http://rhoban.com/robocup2016>

² <http://metabot.cc>

- 2014 (João Pessoa): We took a big step forward by reaching the quarter-finals and working out a robust walk engine.
- 2015 (Heifei): Finally, we coped pretty well with the new artificial grass and colorless field. We reached the semi-finals and took the third place of Kid-Size league.

For this upcoming year, our expectation is to continue to improve the robot's field localization, walk and kick engines by introducing recent advances developed along the year using on our new (strain gauge) foot pressure sensors and learning methods.

This short paper gives an overview of the Rhoban robots hardware and software system in its current state with an emphasis on recent upgrades with the aim to participate to Robocup 2016 in Leipzig.

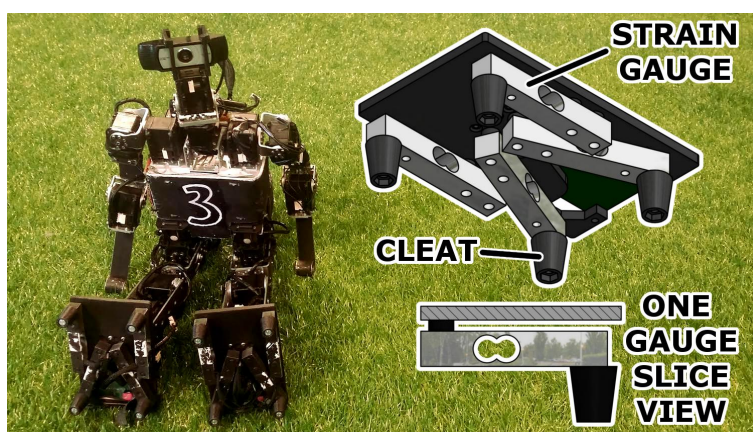
Commitment

The Rhoban Football Club commits to participate in RoboCup 2016 in Leipzig (Germany) and to provide a referee knowledgeable of the rules of the Humanoid League.

2 Hardware Overview

2.1 Mechanical Structure

The mechanical structure of the robot is a classic design using 20 degrees of freedom: 6 for each leg, 3 for each arm, and 2 for the head (pitch and yaw rotations). The global shape of the robot is mainly standard ³.



³ see the robot specification paper

The main novelty of the robot is located in its feet. The feet are no longer flat but are put on the ground on top of 4 cleats at each foot corner. Only these cleats are in contact with the ground and "sink" into the artificial grass. This greatly improve the stability of the robot walking on the new "soft" turf.

In addition to the ground contact, each cleat is linked to a strain gauge force sensor. The whole is integrated into the foot with a piece of electronics and the sensor readings are published on the Dynamixel bus as a virtual device. This low-cost force sensor allows for computing an evaluation of the center of (vertical) pressure for each leg. This new sensor is greatly useful to stabilize the static kick, the walk engine and improve the accuracy of the robot's odometry. All mechanical specifications and electronics design and firmware are available as an open source project ⁴. See [6,7] for a more complete presentation.



Here are the main quantitative values describing the two robots:

⁴ <https://github.com/Rhoban/ForceFoot>

Sigmaban Robot	Value	Unit
Degrees of freedom	20	
Weight	4.5	kg
Height	60	cm
Leg Length	31	cm
Arm Length	27	cm
Foot Length	14	cm
Cleat per foot	4	

Grosban Robot	Value	Unit
Degrees of freedom	20	
Weight	6.5	kg
Height	90	cm
Leg Length	57	cm
Arm Length	32	cm
Foot Length	24	cm
Cleat per foot	4	

2.2 Actuators and Sensors

All the joints are actuated by servomotors. We use off-the-shelf servomotors, that is, Dynamixel RX-28 and Dynamixel RX-64 for Sigmaban and Dynamixel MX-64 and Dynamixel EX-106 for Grosban.

The robot gets feedbacks through the following sensors:

- *Inertial Measurement Unit.* We use a 9 degrees of freedom IMU packaging a accelerometer, a gyroscopic and a magnetometer sensor providing both raws and orientation (yaw, pitch, roll) information through serial communication. The component is a Razor 9-Dof IMU.
- *Camera.* The head of the robot is equipped with a Logitech webcam of type C930 on top of two (pan-tilt) servo-motors. It samples pictures with a resolution of 800x600 pixels with a frequency of about 10 Hz.
- *Joint Positions.* On top of that, the robot uses also joint position feedback provided by each Dynamixel servo.
- *Foot Pressure Sensors.* Each foot has 4 stain gauge sensors integrated with the foot cleats and measuring the applied vertical force. An estimation of the (vertical) pressure point for each foot can be computed. This value is greatly use for stabilization control during walk and kick motion.

2.3 Processing Units

The embedded system is based on two main processing units: a small Cortex ARM7 microcontroller without operating system and a FitPC2i equipped with Linux (Debian 7). The FitPC has 2GB of RAM and is based on 1.6 GHz Intel Atom CPU while the ARM7 has 64kB RAM with 55 MIPS and run at 78MHz. More precisely:

The FitPC is in charge of the high-level behaviour management and the execution of the high-level programmed components:

- *High-level decision processes and behaviours.* The behaviour of the robot is mainly driven by state machines and each different behaviours are implemented into C++ class.
- *Walk motion generation.* The walk generator is splines and inverse-kinematic based. It provides a high level omnidirectional control with forward, lateral and rotation velocities.
- *Motion scheduling.* Communication with low level servo-motors are clocked up to 50 Hz in Linux user space.

- *Vision and localization module up to 10Hz.*
- *Communication with external entities* (via WiFi IP protocol in development environment)

The ARM7 is in charge of the real-time low-level management:

- *Sensors sampling and communication protocol.*
- *Servomotor control.* The processing unit communicates with Dynamixel servos via a serial RS-485 bus.

We now describe in more details some of the above components, in particular the vision module, the localisation module and the motion control system.

3 Vision Module

The vision module of the Rhoban Football Club robot is responsible for making all the necessary image processing and analysis. This module runs on the Linux main embedded computer, is written in C++ and based on the **OpenCV** library (*Open Source Computer Vision Library*) [2]. The whole vision processing is implemented as a pipeline of several connected parametrized filters.

Last year, a huge effort has been done to update the vision detection to the new colorless field environment. Only little improvement will be done on the vision this year since after many tweaks, we were able in Hefei to detect both the goal posts and the ball without major issues.

Our global strategy is to loose the requirement of non false positives for feature detectors in order to simplify the processing and instead rely on the particle filter to select out the likely ones.

The vision module currently has the following characteristics:

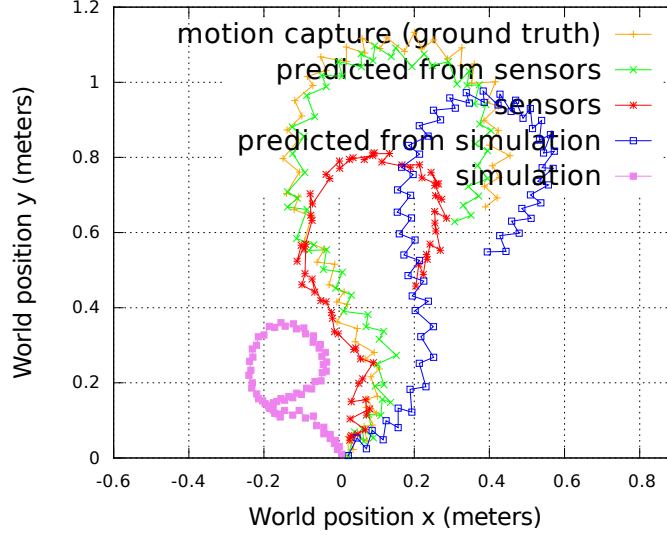
- It essentially uses the raw camera YUV color space.
- An important preprocessing phase firstly extract the field from the image in order to reject the noisy unconstrained environment. Then, the ball and goal are separately looked for.
- Ball detection looks for ball candidates and score them accordingly with their apparent radius, distance and a measure of circularness of shape.
- Goal detector is mainly looking for the goal post bases because they remain in the field green and are easier to extract than upper unconstrained background.

4 Localisation Module and Particle Filters

The localization module allows the robot to know approximately his position on the field. The estimation of the current position is used by the high-level state

machine for taking decisions.

The localization relies on the analysis of the image to find the goal posts and field borders. Two particle filters are running continuously to integrate the detected features. The first filter maintain up to date the ball position with respect to the egocentric robot frame. The second one computes the absolute position and orientation of the robot on the field.



An essential basis for these filters accuracy is the odometry of the robot which integrates the known motion of the robot from internal sensors. The odometry is firstly computed from a complete kinematics model of the robot integrating each footstep using foot pressure sensors and servos feedback position. In order to significantly improve the accuracy and taking account of model discrepancies, ground contact slippery and unknown mechanical backlash, we correct in real time the computed odometry using learning technique (LWPR algorithm) trained with motion capture external ground truth. A complete presentation of this work is detailed in [8].

5 Motions

5.1 Gait Design

The omnidirectional walk motor primitive is mainly open-loop, based on splines and classic inverse kinematic of legs. Three periodic functions defined as three polynomial cubic splines and made of very few points are used as basis for rise, step foot and lateral oscillation movements. Theses spline normalized signals are

then sent to the inverse kinematics X,Y,Z input after some offsets, gains and phase shifts. Angular motor reference positions are then computed for each leg.

There is about 15 "static" parameters such as movement frequency, foot height rise, amplitude of lateral oscillations, etc.. which are not updated during the movement and have to be tuned manually. The 3 other parameters are "dynamic" and used to control the walk (forward, lateral step length and self rotation velocity) from high level.

The whole generator is released as an open source project ⁵ and has been exposed in [7].

6 A New Open-Source Firmware for Dynamixel Servo-Motors



This year, we created an alternative open-source firmware for the MX-64 servo-motor ⁶. Our goal is to get full control over the low-level layer of our robots. More precisely, the default Dynamixel position control could be improved with a more complex internal model of the motor and even more by taking into account the external torque the motor will have to encounter.

We chose the MX-64 for its current sensing capabilities. A custom firmware opens up for a lot of possibilities, such as non-PID control loops, communication protocol optimization and power consumption enhancement. We are currently working on a predictive feedback control approach and a trajectory management. Ideally, instead of sending position orders to every servo-motor we would like to send them a trajectory of near future target positions in order to improve the dynamical accuracy.

Our expectation is to build a new Sigmaban robot using MX-64 servo-motors and custom firmware for next RoboCup competition this summer.

References

1. Rhoban Robots at Yeosu International Expo 2012. <http://rhoban.com/category/arms-2>.
2. G. Bradsky and A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly, 2008.
3. O. Ly, M. Lapeyre, and P.-Y. Oudeyer. Bio-inspired vertebral column, compliance and semi-passive dynamics in a lightweight humanoid robot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'2011)*, 2011.

⁵ <https://github.com/Rhoban/IKWalk>

⁶ <https://github.com/RhobanProject/Dynaban>

4. O. Ly and P.-Y. Oudeyer. Acroban the humanoid: Compliance for stabilization and human interaction. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.
5. O. Ly and P.-Y. Oudeyer. Acroban the humanoid: Playful and compliant physical child-robot interaction. In *ACM SIGGRAPH'2010 Emerging Technologies, Los Angeles*, 2010.
6. Gregoire Passault, Quentin Rouxel, Ludovic Hofer, Steve N'Guyen, and Olivier Ly. Low-cost force sensors for small size humanoid robot. In *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on, (Video contribution)*, pages 1148–1148. IEEE, 2015.
7. Quentin Rouxel, Gregoire Passault, Ludovic Hofer, Steve NGuyen, and Olivier Ly. Rhoban hardware and software open source contributions for robocup humanoids. In *Proceedings of 10th Workshop on Humanoid Soccer Robots, IEEE-RAS Int. Conference on Humanoid Robots, Seoul, Korea*, 2015.
8. Quentin Rouxel, Gregoire Passault, Ludovic Hofer, Steve NGuyen, and Olivier Ly. Learning the odometry on a small humanoid robot. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, accepted.
9. Rhoban Website. <http://rhoban.com>.