RoboCup Junior Soccer Open 2025

Aperture Open

Access to our Linktree and more of our documentation

Abstract

We are Aperture Open, a team of competitors of RoboCup Junior in Soccer Open category, representing Colégio Objetivo from São Paulo city, Brazil. We currently hold the Brazilian first place and Americas third place. This paper summarizes the general hardware and software aspects of our robot, also including a quick overview about our strategy, production process and the implementation of key features for movement, vision and goal scoring.

Leonardo Rosendo (17)
Captain /
Hardware - General

Lucas Suzena (16)
Software - General

Gabriel Hamada (15)
Camera - General

Ivan Pontes (16)
Hardware - Mechanics

Fig. 1 - Robot's photo.

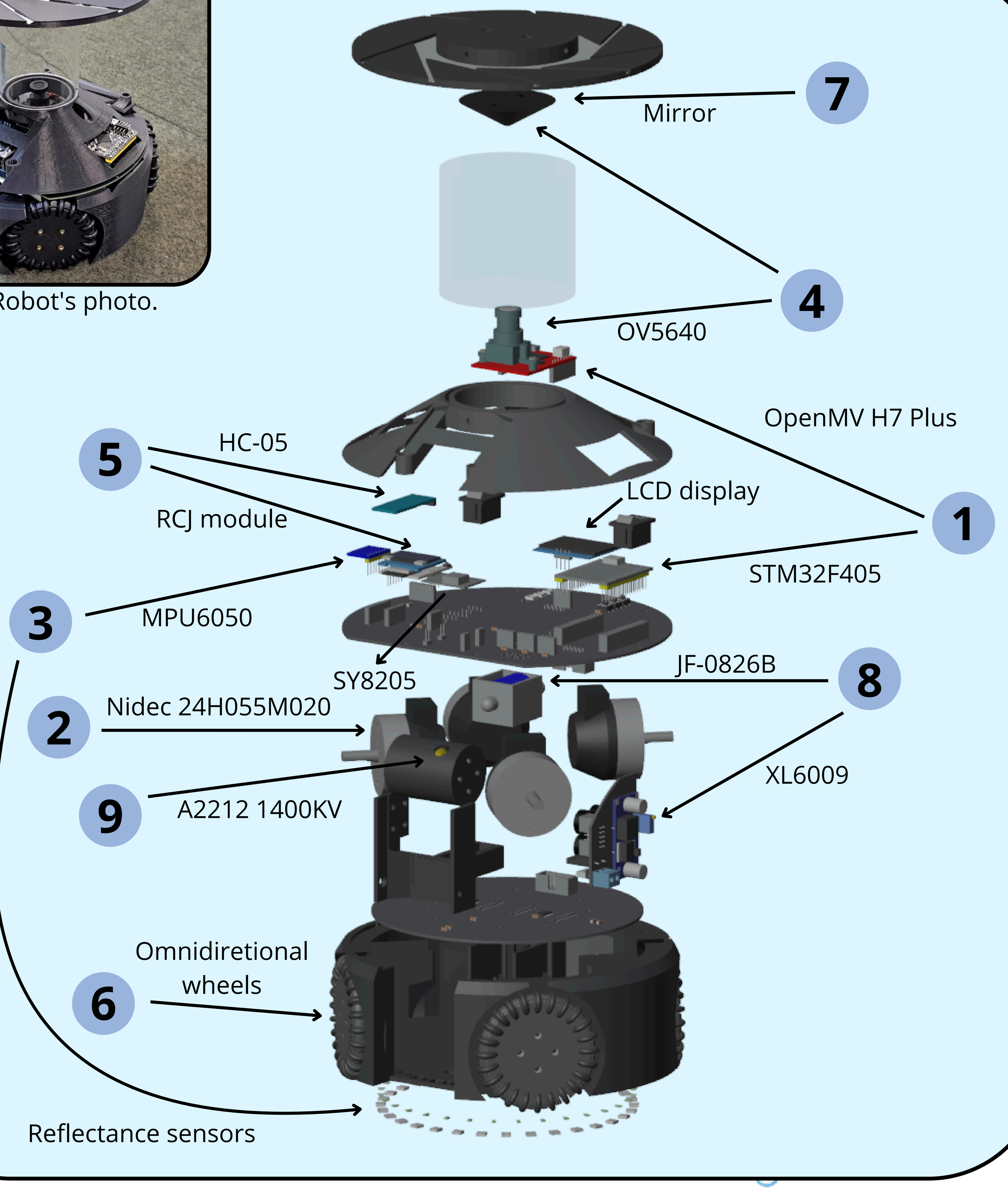

Fig. 2 - Robot's 3D model.

Software

The software of the robot is divided into four parts: Location, Communication, Decision and Movement. Each part plays an important role so the robot can react to the ball and to the field. The programming language used on the robot is C++, using Python only for the camera.

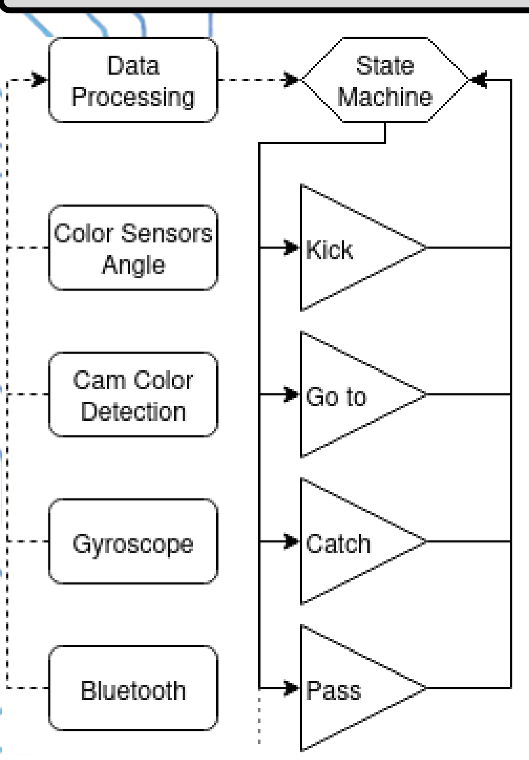

Fig. 9 - Software diagram.

Movement

Moves the motors based on the PID of angle correction and axial movement. The forces in *x*, *y* and own axis for movement are sent by the Decision class. The movement class serves only as an intermediary of the information processing and the physical response of the robot.

Communication

A protocol of serial communication, with the main functions being write and read, to respectively send and receive messages. Those, are lists of bytes with a structure [identifier, index, part 1 message, part 2 message].

Decision

After matches with different teams, we noticed the importance of the strategy when there is a hardware disparity. Differently from the methods of strategy management normally used in Open, we use a method created based in SSL Soccer teams, the State Machine, that uses behaviors, such as striker or goalkeeper, and various states, such as kicking or going to a position. They are controled by transitions, and allow a strategy much more fluid and changeable, where we can define how the robot will react based on the analysis made of the team we are playing against.

Location

Responsible for the sensors reading, getting the angle of the robot in relation to an offset and which direction is the robot on a line.

The camera uses a color-tracking system to identify the goals and the ball. With the absolute position of the goals on the field and relative angles to the robot, we use trigonometry to find the position of the robot on the field, and with it the position of the ball. We program the camera using the OpenMV firmware.

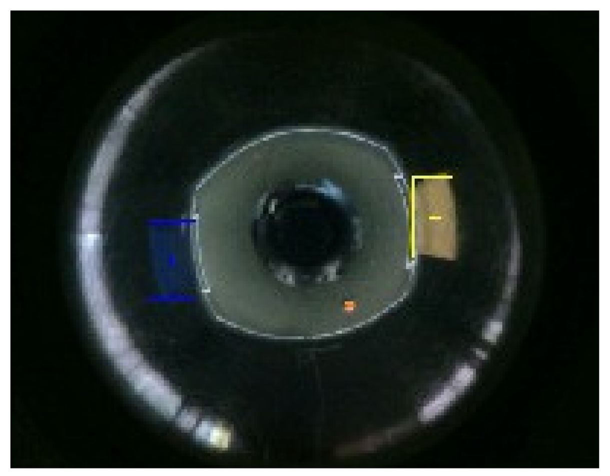

Fig. 10 - Camera image on OpenMV.

Example Strategy

Striker

The striker focuses on getting the ball and finding the best path to score considering the current situation. It can also change to defense positioning if necessary.

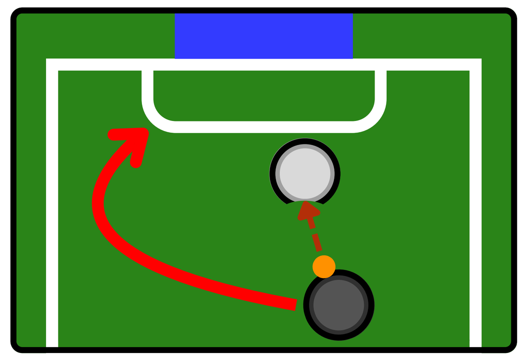

Fig. 11 - Striker's ideal path to score.

Goalkeeper

The goalkeeper goes to the best place to protect considering the ball position, but can change to attack assistance if close to the ball.

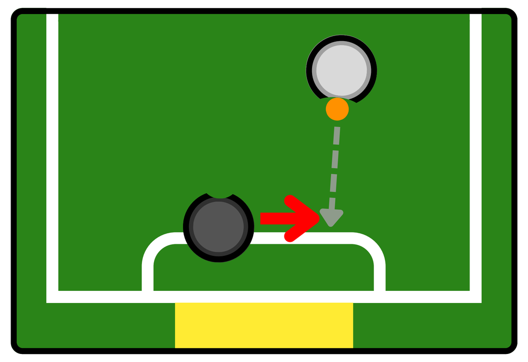

Fig. 12 - Goalkeeper defense.

Hardware

The robot has three boards. The top board has 17 cm in diameter and is where most of the components are. The bottom board has 14 cm in diameter and is focused on the reading of all the reflectance sensors using two multiplexers, each one responsible for 16 sensors. Finally, the kicker board is where the voltage is increased for the kicking system. The main hardware functions are detailed below, while the exact components can be seen on the left.

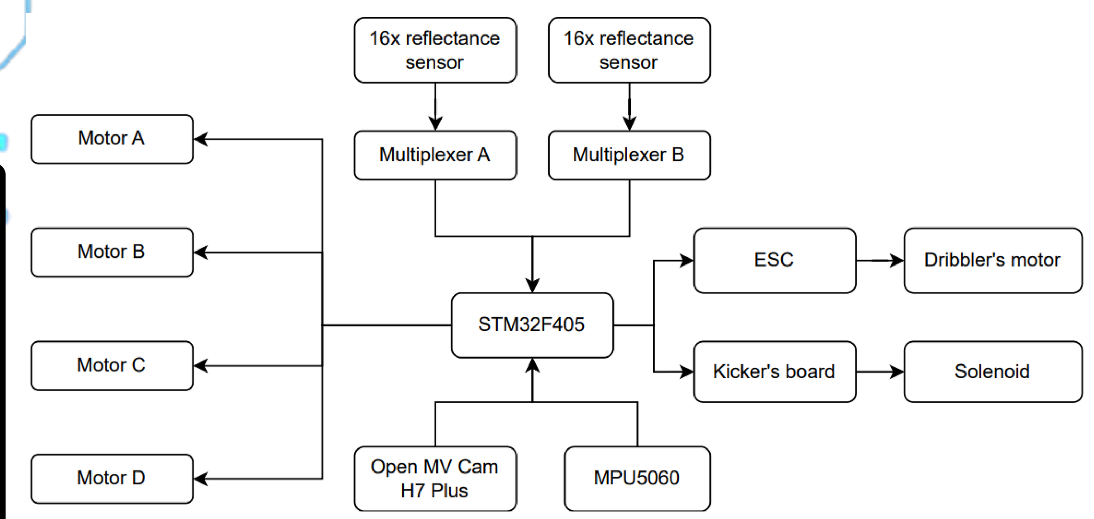

Fig. 3 - Hardware diagram.

1 Processing

On the middle board there is a STM32F405 that reads all the sensors, controls the movement and sends informations to the other robot. On the top board there is an OpenMV H7 Plus, that is responsible for the part of the code related to the camera.

2 Motors

The four motors of model Nidec 24H055M020 of 12V and 3900 RPM are responsible for the movement of the robot and are disposed perpendicularly to each other. They have an integraded encoder, and their speed and turn direction are controlled by the main microcontroller

3 Detection

The robot uses a MPU6050 gyroscope and the camera to get the angle of the robot in relation to an offset and 32 reflectance sensors, each one made by one TEMT6000X01 phototransistor and one WS2812B addresable LED, to detect the white lines of the field.

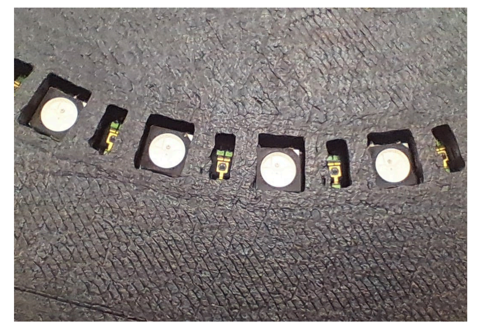

Fig. 4 - Reflectance sensors on the bottom board.

4 Vision

The OV5640 camera is positioned pointing at a conical mirror to achieve a 360° view of the field, allowing the detection of the ball and goal colors. Using their distance and angle the robot and ball positions are calculated.

5 Communication

The robot's communication is divided into two parts: bluetooth communication between robots with an HC-05 or RCJ bluetooth module; and communication via serial between microcontrollers and between boards, with a flat cable.

6 Omnidirecional-wheels

The wheels of the robot are omnidirectional-wheels, which allows movement to all directions. They are made with 3D printed parts in ABS and a total of 20 sub-wheels. The total diameter of the wheel is 6.5 cm.

Fig. 5 - Omnidirecional-wheels.

7 Mirror

The conical mirror production proved to be very difficult because of the lack of materials available to buy in Brazil. We ended up producing the cone with an aluminum cylinder on a lathe, that was later polished to get the ideal reflection.

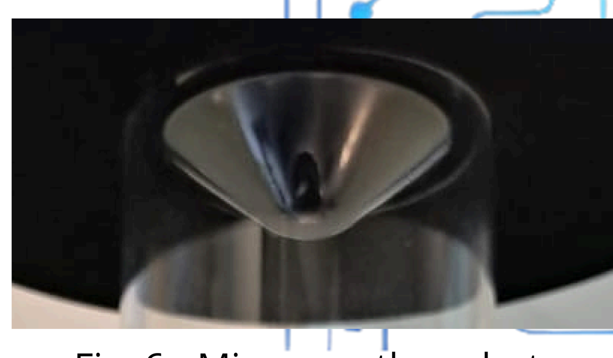

Fig. 6 - Mirror on the robot.

8 Kicker

We use a solenoid of 6V and 20N, that is powered with the 48V converted by our step-up. This provides us with a stronger but smaller kicker, that is activated using a mosfet. This system is located on its own board.

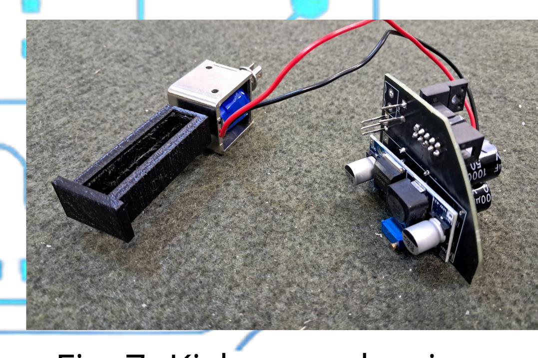

Fig. 7- Kicker mechanism.

9 Dribbler

Using 35% of its power, the dribbler motor goes up to 6200 RPM Because of the lack of space on the robot, the case of the motor is used as the dribbler roller, that has parts molded in silicone to provide grip.

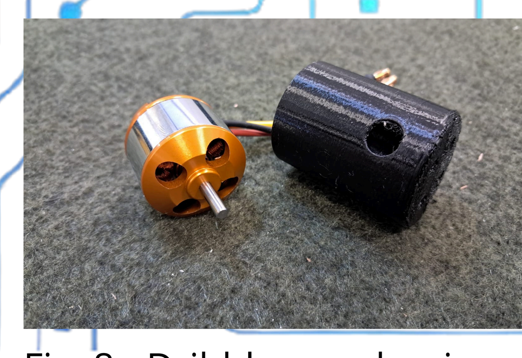

Fig. 8 - Dribbler mechanism.

Production

The boards of the robot were designed using EasyEDA and printed in copper and glass fiber by JLCPCBs. The supports were designed in OpenSCAD and 3D printed in PLA and ABS. We also use an acrylic tube to support the mirror structure without obstructing its vision.