

JdeRobot-Kids framework for teaching robotics and vision algorithms

Julio Vega, J.M. Cañas, Miguel Cazorla
 julio.vega@urjc.es, jmplaza@gysc.es, miguel.cazorla@ua.es



Introduction

Secondary Education has been a complex and problematic educational level in Spain; by the age covered, its intermediate character between Primary and University. Furthermore, it exists the incentive demanded by today's society to incorporate New Technologies in the classroom, the result is a rather chaotic educational system.

Here we introduce the implementation in the classroom of a new pedagogical methodology, constructivism, using the software and hardware platform we have developed: JdeRobot-Kids and PiBot.

Results

JdeRobot-Kids has been tested in different high-schools with very good results. A summary of the student surveys is shown below:

- >70 % Rate Robotics as very interesting subject (score between 8-10).
- >60 % Rate the documentation received between 8-10: the JdeRobot-Kids manual.
- <40 % Rate JdeRobot-Kids between 5-7.
- >70 % Rate the practices very interesting (8-10).

With all the above, the overall assessment given to the JdeRobot-Kids course is included in the next figure.

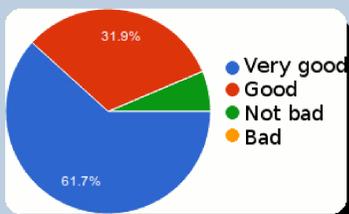


Figura 1: Overall rating of JdeRobot-Kids

Conclusions

According to our experience in Spanish educational centers, and on the constructivist methodological bases of J. Piaget, we have designed and implemented a complete educational proposal, developing a middleware (JdeRobot-Kids) which supports a multi-platform robotic hardware (Arduino standard, mBot, simulated mBot, and PiBot), as well as a complete teaching program, all of them adapted to the educational needs of the pre-university students. The results have been very satisfactory. The surveys show a great satisfaction by students, as well as the level of use and learning in the classes.

References

- [1] Julio Vega and J.M. Cañas. Curso de robótica en educación secundaria usando constructivismo pedagógico. *JITICE 4th Workshop, Educational Innovation and ICT, U. Rey Juan Carlos, November 26th 2014. ISSN: 2172-6620, 2014.*
- [2] Julio Vega and J.M. Cañas. Entorno docente con arduino y python para educación robótica en secundaria. *JITICE 5th Workshop, Educational Innovation and ICT, U. Rey Juan Carlos, October 25th 2016. ISBN: 978-84-697-0892-7, 2016.*

Teaching environment: JdeRobot-Kids

The teaching environment we have developed, JdeRobot-Kids^a includes the software middleware, coded in Python language, which supports three robotic hardware platforms: Arduino standard, mBot, simulated mBot, and the *Raspberry Pi3* based robot PiBot, as well as a complete educational program for pre-university students.

Arduino hardware was chosen as a robotic hardware because it is free hardware and there are some educational pre-assembled kits such as mBot. Both of them, Arduino standard board and mBot, are supported by the middleware. Furthermore, a new robotic platform has been developed, the PiBot (2), based on another multipurpose free-hardware board: Raspberry Pi3. It has been mounted over a iron chassis with wheels and steered with two *Servo Feedback 360° by Parallax*. It is controlled thanks to the GPIO ports of the board. A *PiCam* camera has been mounted over a servo pan-tilt unit and is supported with a driver, *PiCamServer*, which is included in the middleware to ease use for students.

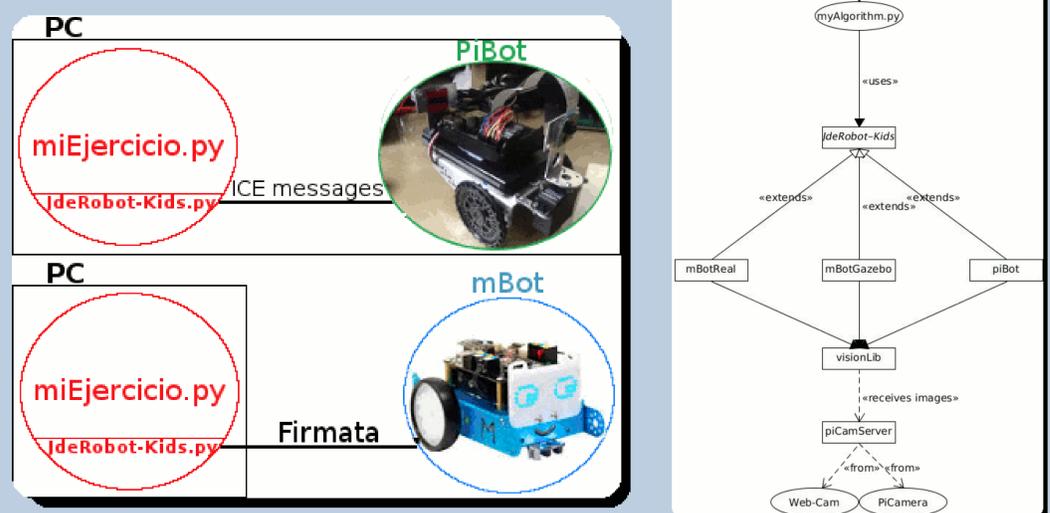


Figura 2: JdeRobot-Kids supports mBot and PiBot; and use diagram

The middleware establishes connections depending on the robot is being used, as it is shown in the use diagram above.

To summarize, the main bases of the environment are:

- Multiplatform support: Arduino, mBot and PiBot (Raspberry Pi 3).
- PiCam: camera supported under PiBot to teach basic **vision algorithms**.
- Programming language: **Python**, as a plausible language to learn about programming Robotics.
- Collection of several **robotics programming practices** for a complete academic course.

^a<http://jderobot.org/JdeRobot-kids>

Using the middleware

The students just need to coded the algorithm in a Python file as it can be seen in the example code below, using the functions provided by the middleware. The main methods are shown in the next figure.

```
import JdeRobotKids

if __name__ == "__main__":
    myJdeRobotKids = init()

    vel = 100
    while True:

        val = myJdeRobotKids.leerUltrasonido()
        print(val)
        if (val < 10):
            myJdeRobotKids.retroceder(vel)
            time.sleep(0.5)
            myJdeRobotKids.girarDerecha(vel)
            time.sleep(0.3)
            myJdeRobotKids.avanzar(vel)
            time.sleep(0.3)
```

| Actuation | Sensing |
|--------------------|----------------------|
| move (V, W) | leerIntensidadLuz |
| avanzar (V) | leerUltrasonido |
| retroceder (V) | leerIntensidadSonido |
| girarIzquierda (W) | leerPotencimetro |
| girarDerecha (W) | leerMandoIR |
| parar | |
| encenderLed | |
| apagarLed | |
| escribirTexto (T) | |
| playTono (Fs) | |

Figura 3: Python example code using JdeRobot-Kids with its different provided functions