Smartphone Robot for High School Students: RobHiSS

Bruno Martins, André Costa, Carlos Caetano,
Carlos Rodrigues, Gonçalo Ruão, Isaac Lopes,
João Aguiar, Pedro Sousa, Pedro Silva, Tiago Correia

Faculty of Engineering, University of Porto

Armando Sousa

INESC TEC (formerly INESC Porto)
Faculty of Engineering, University of Porto
Email: asousa@fe.up.pt

***Abstract* — This project arose from the absence in the market of a modular smartphone controlled robot capable of encouraging high school students to program and apply the physics and math's knowledge learned into it. Therefore this project's intention was to study the best way to develop a do-it-yourself (DIY) cost effective robot using only components off the shelf (COTS) and benefit from the omnipresence of smartphones. With the objective of making this robot attractive to anyone with low programming skills, it was important to make it configurable in an easy to understand language and a simple user interface, like the ones provided by Scratch and the MIT AppInventor2. The functional, physical and non-functional requirements for this robot and the free software developed are presented and validated attesting that this project was successfully completed.**

I. Introduction

This paper presents and describes the highlights of a project to build an educational robot that will encourage kids and high school students to learn how to program and apply maths and physics to a modular robot with a smartphone on it.

As a guideline to its development, it needed to be able of participating in RoboCupJunior Rescue B which imposed some limits on the technology applied, dimensions and functionalities.

The assembled team was divided into three main groups which were responsible for the three main sections of this robot. Each one of the sub-teams has researched the best way to achieve their objectives taking into consideration the cost and time needed to implement the different possible solutions.

This project's goal is set on building a modular autonomous line follower robot capable of being programmable by high school students into different configurations and functionalities, thus stimulating their interest in learning a programming language, understand the basic laws of physics and apply maths to the algorithms created.

II. Requirements

The team had into consideration the RoboCupJunior Rescue B rules and requirements as well as the ones imposed by the client on the several meetings that took place

A. RoboCupJunior Rescue B Rules Summarize

Some of the most significant limitations imposed were:

- Having more than 30 cm of width
- Looking over 30 cm of height
- Wireless communications (except for Bluetooth class 2 or 3 and ZigBee)
- Pre-mapped type of dead-reckoning
- The use of lasers
- Capable of lifting up an empty 33cL can
- Detect the rise of the air temperature by at least 10 Celsius degrees and signalling it

B. Requirements and restrictions

The client has also established some objectives and main restrictions for the robot:

- It has to be inexpensive with an maximum cost of 100 €
- The main processing unit has to be a smartphone and take advantage of its camera and other functionalities
- It has to be modular and easily re-programmable
- It has to be capable of following a coloured line
- It has to be capable of encouraging high-school students to learn how to program and apply both physics and maths on a robot
- It has to be easy to build with COTS
- It has to safely stop moving upon bumping into anything

After analysing the requirements from all the stakeholders, they were grouped into three main categories:

1) Functional requirements:

- The ability of following a coloured line on the ground
- The ability of detecting air temperature variations
- The ability to pick up an empty can
- The existence of an on/off button
- Immunity to interferences (such as camera flashes)

2) Physical requirements:

- The robot shall move using wheels
- The ability to proceed along bumps and small debris
- Light and moderately fast
- Easy to assemble and rebuild with easily obtained COTS

3) Non-functional requirements:

- Educationally valuable
- The existence of a construction manual
- The existence of commercial documentation

Given the above requirements, the challenge was developing new ideas and solutions while, as intended by RoboCup Rescue project, promoting this socially significant domain and enhance team member's experience in engineering projects.

III. MARKET RESEARCH

With its goal set, the team studied how to develop a robot capable of receiving controlling instructions from a smartphone to be interpreted on a microprocessor.

Therefore the project started to take shape, and the next stage was to decide the functionalities to be implemented while satisfying the imposed restrictions. Hence it was decided to build a small robot in a sturdy material with two traction wheels, a stand to fit the smartphone, a claw in the front to lift an empty can, a battery pack as the power source placed in the back for stability, one temperature sensor and one buzzer to announce the rise on the temperature above a threshold and two interruption switches that upon contact would open the electrical circuit that supplies the motors. With these guidelines two computer assisted designs were made and are shown in figure 1.

Figure 1. Robot's initial computer aided designs

At this stage the team divided the robot into three main sections to research and develop:

1) *Structure and Locomotion* which groups the base, the claw system, the smartphone stand, the interruption switches, the motors and the wheels.
2) *Communication and Interfaces* responsible for the communication's protocol and components, the temperature sensor, the buzzer and the power source.
3) *Processing and Control* designated to the software development and the selection of the microprocessor.

IV. PROPOSED APPROACH: ROBHISS

The development stage was next and a functional block diagram was made to design the system's behaviour and interactions with itself and the outside world. As shown in figure 2 the system combines a smartphone with all the necessary components to implement the required functionalities.

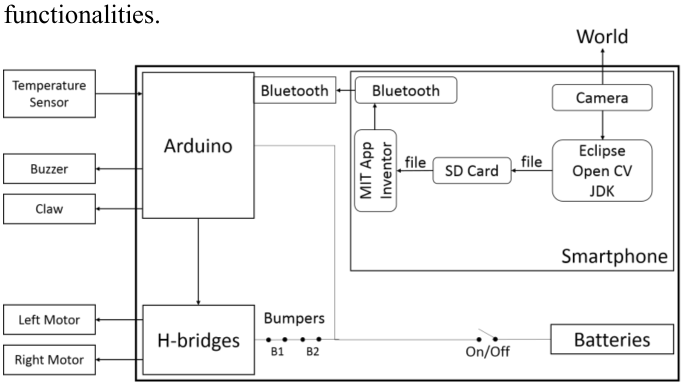

Figure 2. System's function block diagram

Two applications were developed for the smartphone. One for image processing, programmed in Eclipse using OpenCV libraries, to calculate the distance from the centre of the line to the middle of the screen and the angle made with the screen's bottom, writing them on the SD card. This application's algorithm is detailed in figure 3 and the parameters' values used to determine if a pixel is similar to the selected one are presented on table I, where the hue value is relative to the selected pixel and the other values are absolute.

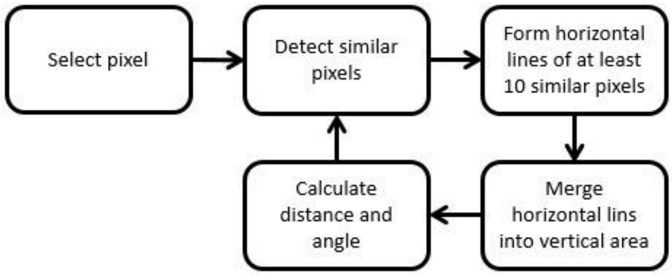

Figure 3. Image processing algorithm strategy

TABLE I. IMAGE PROCESSING PARAMETERS

Parameters	Minimum Value	Maximum Value
Hue	-15	+15
Saturation	60	255
Luminosity	60	255

The second application was developed using the MIT App Inventor 2 open source IDE to convert the values written by the image processing app into instructions interpreted by arduino's PWM outputs (from 0 to 255) to be transmitted in an array via bluetooth. A communication protocol was defined and implemented to standardise the messages sent to the arduino. Its structure can be seen in table II.

TABLE II. BLUETOOTH MESSAGE COMPOSITION

Position	0	1	2	3	4	5	6
Value	$	# id	,	# Left	;	# Right	&

The "$", ",", ";" and "&" characters are delimiters of the message, while the three remaining characters contain the message's id and both control values. The id is used to determine their arrival order and detecting faulty communications. The control values can go from 0 to 255 or assume other specific values with their function described on table III.

TABLE III. BLUETOOTH MESSAGE'S CONTROL VALUES

Function	Left Value	Right Value
Move backward	0 to 127	0 to 127
Move forward	128 to 255	128 to 255
Pick up can	300	300
Lay down can	301	301
Set new temperature threshold	301	threshold

The temperature variation functionality was designed to be interpreted directly by the arduino avoiding the need of transmitting this information to and from the smartphone.

Building a prototype was necessary to test and validate the designed product and, thus, an objective model with all the desired functionalities was developed - the RobHiSS.

Two phases were proposed, one being the construction of the structure and the printed circuit board (PCB) while the other one being the assembly of all components.

Main components:

1) *Base structure* with 16 x 20 x 4 cm dimensions was cut from a roofmate board because of its resistance and easiness to work with.
2) *Claw* to lift an empty can was designed obeying to a relationship between resistance and weight. Thus it was built according to the scheme presented in figure 4, from a polyvinyl chloride sheet (PVC).

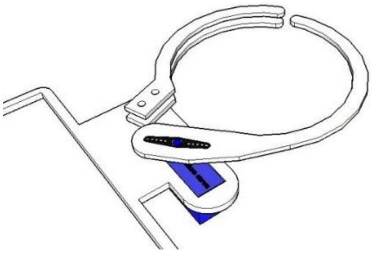

Figure 4. Proposed robot's claw design

3) *Printed Circuit Board (PCB)* to take the prototype a little further simplifying the components' assembly and enhance the circuit's reliability. It can be seen in figure 5 with some of the main components already attached.
4)

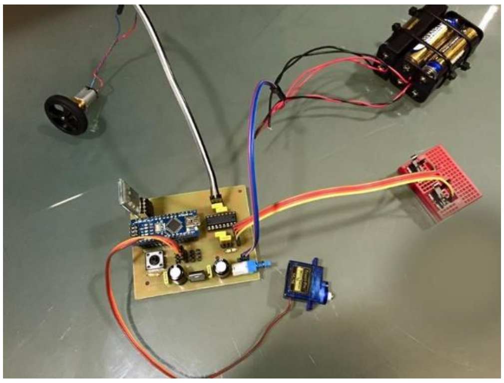

Figure 5. PCB with some of the main components

5) *Arduino nano* responsible for interpreting the commands sent from the smartphone, the temperature variation and operating the actuators accordingly.
6) *Gearmotors* to provide the differential locomotion.
7) *Servomotors*, one of them is responsible for the grab and release actions of the claw while the remaining two lift and lower the claw itself.
8) *Bluetooth module* responsible for implementing the communication's physical layer.
9) *AA batteries* equivalent to 7.2-9 nominal DC volts which work as the power supply of the whole system.

A complete list of materials is shown on table IV with the required quantities and prices.

TABLE IV. LIST OF MATERIALS

Components	Qty.	Local Prices	Online Prices
Arduino Nano	1	38.67 €	2.58 €
Bluetooth Module	1	17.75 €	3.54 €
H Bridges L293DNE	1	3.08 €	0.85 €
Gearmotors and wheels	2	35.87 €	5.14 €
Third wheel (roll-on)	1	- €	- €
180 degrees servomotors	3	17.85 €	2.58 €
Temperature sensor MCP9700	1	0.26 €	0.26 €
Buzzer	1	11.60 €	8.99 €
Microswitches	2	0.76 €	0.76 €
Power switch SPDT	1	0.35 €	0.35 €
Voltage regulator UA7805C	1	0.72 €	0.72 €
AA batteries	6	2.57 €	2.57 €
Two batteries sockets	3	3.78 €	1.40 €
Breadboard	1	2.68 €	1.33 €
PVC sheet	1	- €	- €
Roofmate board	1	- €	- €
Copper wires, glue, sponge	qs	5.00 €	5.00 €
Totals	27	140.94 €	36.07 €

All these components were developed or acquired and then assembled, and the final result can be seen in figure 7.

The interface of the application developed with the MIT App Inventor 2 IDE can also be seen in figure 6, containing a select button to choose the RobHiSS's bluetooth module, start and stop communication buttons and an image processing button to launch the application responsible for it. After that, the user chooses the colour of the line for the robot to follow by clicking on the screen and it will start moving.

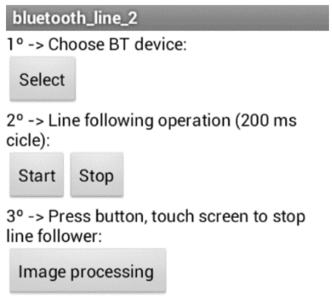

Figure 6. MIT App Inventor 2 application's interface

For the sake of assisting anyone attempting to build the RobHiSS and in order to display the developed work, a website was created containing general information about the project and team, demonstration videos and a construction manual, which can be accessed through the following url http://paginas.fe.up.pt/~ee08114/SEAI/index.html.

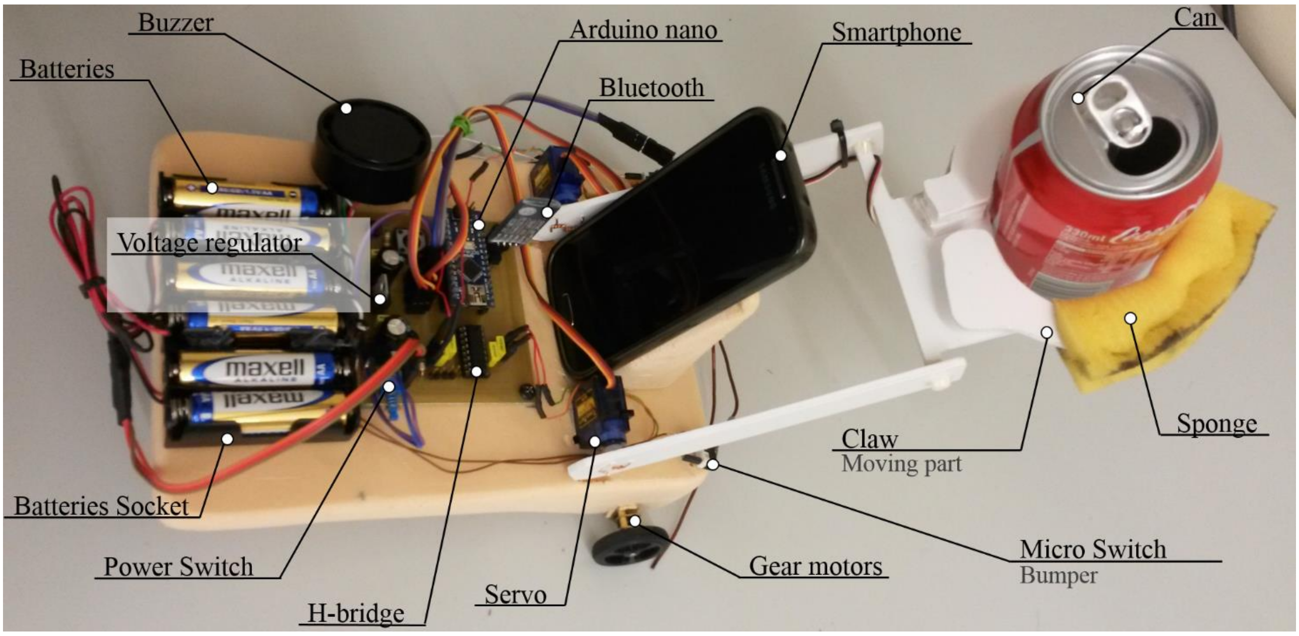

Figure 7. Final prototype

V. RESULTS VALIDATION

To prove that the achieved solution complies with what it was made for, the requirements were tested. These results can be seen on the tables V, VI and VII divided in functional, physical and non-functional requirements, respectively.

TABLE V. FUNCTIONAL REQUIREMENTS VERIFICATION

Requirements	Verification Method	Result
Ability to follow coloured lines	Place the robot on top a coloured line and start it	Passed
Ability to detect a temperature increase above 10º C	Aim a hair dryer set to heat to the robot	Passed
Ability to lift an empty can	Execute grand and lift routines	Passed
Stopping safely if it enters in contact with anything	Test the effectiveness of the microswitches	Passed
Immunity to external interferences	Place the robot close to microwaves, TVs and phones	Passed

TABLE VI. PHYSICAL REQUIREMENTS VERIFICATION

Requirements	Verification Method	Result
Ability to move correctly with two wheels	Make the robot move straight forward for at least 2 meters	Passed
Not being affected by bumps and debris	Place some bumps and small debris on the robot's route	Passed
Light and moderately fast	Weight and measure its speed	Passed

TABLE VII. PHYSICAL REQUIREMENTS VERIFICATION

Requirements	Verification Method	Result
The robot's components cost must be below 100 €	Verify the sum of the cost of all components	Passed
Layered software architecture and debris	Assess software architecture debris on the robot's route	Passed
Light and moderately fast	Weight and measure its speed	Passed

VI. FUTURE WORK

The forthcoming work involves the design of custom made kits for the clients to assemble in partnership with electronic stores and the development of more applications for the robot.

ACKNOWLEDGMENT

This work is partially financed by the ERDF European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project FCOMP-01-0124-FEDER-037281. The authors also wish to thank the Dept. of Electrical and Computer Engineering of FEUP for another part of the financing of this work.

REFERENCES

[1] T. M. Examiner, T. Faculty, and E. Engineering, "Inter Process Communication in Android," no. March, 2012.

[2] R. Meier, Professional Android 4 Application Development. 2012.

[3] R. K. Sure and S. Patil, "Android Based Autonomous Coloured Line Follower Robot," pp. 368–373, 2014.

[4] A. Fosdick, "PCB Design with Altium : PCB Layout and Ordering a Board," pp. 1–31, 2011.

[5] C. Nimodia and H. Deshmukh, "Android Operating System," Software Engineering, ISSN, vol. 3, no. 5, pp. 10–13, 2012.

[6] L. F. Robot, "Line Following Robot," pp. 1–11, 2008.

[7] T. S. D. Siqueira, "Bluetooth - Caracter´ısticas, protocolos e funcionamento," Universidade Estadual de Campinas, 2006.

[8] E. Pontif, U. Cat, and R. Grande, "Bluetooth technology (ieee 802.15)," Group.

[9] K. Dawson-Howe, A Practical Introduction to Computer Vision with OpenCV (Wiley-IS&T Series in Imaging Science and Technology). 2014.

[10] G. Bradski and A. Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library. O'Reilly Media, Incorporated, 2008.

[11] A. Chopra, Google SketchUp 7 for Dummies. 2009.

[12] Wei-Meng Lee, Beginning Android 4 Application Development. 2012.