© 2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works

Link to publisher version: https://doi.org/10.1109/TLT.2023.3244604

Robobo SmartCity: An Autonomous Driving Model for Computational Intelligence Learning through Educational Robotics

M. Naya-Varela, *Member, IEEE,* S. Guerreiro-Santalla, T. Baamonde, F. Bellas, *Member, IEEE*

***Abstract*—This paper presents the Robobo SmartCity model, an educational resource to introduce students in Computational Intelligence (CI) topics using educational robotics as the core learning technology. Robobo SmartCity allows educators to train learners in Artificial Intelligence (AI) fundamentals from a feasible and practical perspective, following the recommendations of digital education plans to introduce AI at all educational levels. This resource is based on the Robobo educational robot and an autonomous driving setup. It is made up of a city mockup, simulation models, and programming libraries adapted to the students' skill level. In it, students can be trained in CI topics that support robot autonomy, as computer vision, machine learning, or human-robot interaction, while developing solutions in the motivating and challenging scope of autonomous driving. The main details of this open resource are provided with a set of possible challenges to be faced in it. They are organized in terms of the educational level and students' skills. The resource has been mainly tested with secondary and high school students, obtaining successful learning outcomes, presented here to inspire other teachers in taking advantage of this learning technology in their classes.**

***Index Terms*—Computational intelligence, Educational robots, Educational simulations, Machine learning, Mobile and personal devices, Robot programming, STEM**

I. Introduction

THE impressive advances of Artificial Intelligence (AI) in the last two decades are increasingly affecting the society as a whole. Intelligent algorithms are now common in real applications like recommendation systems, autonomous cars, smart apps, domestic robots, etc. Current and future generations will be deeply impacted by intelligent systems in their everyday tasks and jobs [1][2]. Hence, it is essential that current students receive a formal education about AI not only to face, but also to understand, all these challenges to come. This necessity has prompted policy makers worldwide to encourage introducing AI topics in the official curricula at different educational levels [3][4].

The goal for the educational community is to develop formal teaching resources that can be used in the short-term at classes to support such curricula [5]. There are many valid approaches to face AI education, from more theoretical to more practical, depending on the specific audience and educational level. But all of them rely on the existence of suitable and feasible materials, like computer applications, simulations, unplugged activities, real devices, and others. In this scope, robots are optimal educational tools to train students in AI topics. From a methodological perspective, using robots in classes promotes learning by doing methodologies, interdisciplinary training, cooperative learning, and project-based learning [6][7][8]. From a literacy point of view, they allow learning about perception, actuation, machine learning, reasoning, or representation [9]. In addition, they introduce students to the problem of dealing with the real world, where usually algorithms do not work as expected, due to unexpected technical drawbacks, miscalibrated sensors or actuators, lack of proper illumination, etc. Finally, from the economic perspective, equipping educational centers with basic educational robots is not highly expensive. Many of them, worldwide, already have real robots that could be used, and others could rely on open simulation platforms as a latter option.

The current work aims to contribute to the advance on AI education by *presenting a robotic open learning resource that can be directly used by teachers, or adapted to their needs, to train students in the fundamentals of AI from a practical perspective*. It is based on the *Robobo* educational robot [10], widely tested for AI learning at university and pre-university levels [11][9], and in an autonomous driving model, called the *Robobo SmartCity*. It is made up of a city mockup, simulation models, and programming libraries adapted to the students' skill level. We have followed an *Educational Design Research methodology*. Firstly, designing the model to face the urgent requirements in education about AI. Secondly, developing its main components (e.g. libraries, simulators, etc.), and finally evaluating educational interventions with the model that were used to improve it in subsequent stages.

In the design stage it was decided to highlight the computer science aspects of robotics as compared to the "mechatronic" ones. That is, the model faces *Computational Intelligence (CI)* learning, understood as all those computer science techniques

All authors are with the Integrated Group for Engineering Research of the CITIC Research Center (University of Coruña), Spain (e-mail: martin.naya@udc.es), (e-mail:sara.guerreiro@udc.es), (e-mail: tamara.bardao@udc.es), (e-mail: francisco.bellas@udc.es).

that support the robot autonomy, leaving manipulative skills for other subjects or courses. In a second step, it was addressed a *general organization of topics, skills, and robot type* for CI learning at different education levels, from primary school to university degrees. In this way, we started from a formal and global perspective on the utilization of robots in education. Thus, it was decided to narrow the coverage of the research and focus it on *secondary school and high school* education. Thanks to that, we could analyze the model validity in pre-university levels, where CI topics are novel, and the impact could be higher. Specific challenges and teaching units were proposed and tested for these educational levels, in many training sessions with students using the model. The results encompass the research carried out in the last 6 years.

The remaining of the paper has been structured as follows: Section II will be devoted to presenting the theoretical basis of this approach, based on educational robotics, AI, and the requirements of the suitable robots in this scope. The related work will be described in section III, highlighting how they differ from the current proposal. Section IV describes the research methodology, including: the research questions that have been faced, a tentative organization of CI learning stages with robots, the description of the Robobo Project and the Robobo SmartCity model, and finally, the data collection instruments and analysis procedures. Section V presents the results of the model utilization with secondary and high school students, analyzing the validity of the proposal and answering the research questions. Finally, section V contains a discussion about the positive and negative aspects of this approach and how it can advance AI education.

II. Theoretical Basis

A. Educational robotics

Educational robotics has become increasingly popular at different teaching levels in the last decade, mainly pre-university ones [12][13][6]. The use of educational robots at the beginning of this century was mainly limited to specialized university degrees like electronics, automation, or computer science. However, the arousal of open source and low-cost hardware platforms, together with the development of programming languages suitable for younger students, implied that robots could be introduced in secondary, and even primary, schools [14][7][14]. In the last years, many different approaches in the use of robots in pre-university education have been proposed in the form of specific curricular subjects [16][8][17], extra-curricular activities [18][19], or competitions [20][21].

Currently, we can find robotics' subjects in primary and secondary school focused on teaching programming fundamentals [22], others on manual training using construction kits to build robots by the students [23][24], and others that use the robots integrated into STEM (Science Technology Engineering Mathematics) courses [13][25] to take advantage of their interdisciplinary possibilities. However, as commented above, current digital education plans promote going beyond these "classical" applications and using robots as tools to teach Artificial Intelligence (AI) from a practical perspective [26][27]. They make up a feasible learning technology that can be implemented in the short-term at different education levels. The question that arises is how to do it in a structured and formal way.

B. Computational Intelligence Education

Developing AI literacy for different educational levels is an active and challenging research area [27][28]. On the one hand, we can find remarkable worldwide initiatives [29][30], focused on creating formal learning resources for topics like perception, learning, reasoning, or natural interaction, adapted to the student's age and skills. All of them highlight the importance of providing students, mainly in pre-university levels, with a "specific view of AI" focused on solving problems in real devices, escaping from more theoretical and classical views inherited from university courses. This is also the approach of the most relevant textbooks in the field, as [31], where AI systems are treated as *intelligent agents*. These agents are situated in an environment, perceiving it, and performing actions to fulfill their objectives autonomously. As it can be observed, the intelligent agent approach fits perfectly with using robots to learn AI.

Regarding the Computational Intelligence (CI) approach proposed here, more focused on computer science aspects of robotics, it fits with the most advanced curricula already developed and tested [32][33]. These initiatives include teaching units to train students on the fundamentals of computer vision, natural interaction, or machine learning, as well as acquiring other skills to support such topics, as computational thinking or problem-solving. Furthermore, as these approaches follow the intelligent agent paradigm, they also include other aspects related to real systems, like sensing, actuation, and basic control.

C. Robotic Platform Requirements

In addition to the curriculum content that has to be generated for each educational stage, it is required to have robotic platforms that support CI training at classes. The following features must be considered for them:

a) *Low-level sensorial capacity:* to introduce students to intelligent robotics they must first acquire basic knowledge about robotics. To this end, robots should be equipped with sensors that support basic operations, like navigation or object manipulation. These types of sensors provide distance, orientation, light intensity, simple line detection, or contact.

b) *High-level sensorial capacity:* to obtain suitable training for 21st century AI and digital skills, it is necessary to have sensors that allow natural interaction with humans and the environment, such as cameras, microphones, speakers, and tactile sensors. In this way, projects that involve seeing, hearing, and detecting contact can be proposed, promoting Human-Robot Interaction (HRI).

c) *Complete set of action/interaction capabilities:* the greater the capacity to act in the real world, the longer the lifetime of the robot in the classroom because a greater number of different challenges can be carried out. Thus, similarly to the robot sensing capacity, it is necessary to provide it from basic actuators such as those associated with locomotion (wheels, legs) or manipulation (arms, hooks, tweezers), to high-level actuators, more related to HRI like loudspeakers (to produce speech) or high-resolution screens (to show visual information and emotions).

d) *Permanent communications:* future intelligent robots will be connected into a collective system with other AI systems and to the internet. With this feature, challenges involving taking remote information or other involving multi-robot systems can be developed with students.

e) *High computing capacity:* devices with high computing capacity are required to process information from real-world sensors such as the camera or the microphone, produce speech, real-time information transmission, and execute CI algorithms. This way, such processing can be performed in the own robot, without relying on external computers which reduces its autonomy.

f) *Low cost:* as a key requirement for pre-university education, hardware devices used in robotics teaching should be inexpensive, since the number of students is high, and the investment capabilities of the schools and education centers are moderate.

g) *Long-term usability:* in the current technological market of electronic devices such as smartphones or cameras, new devices with new features are launched every day. However, formal education should depend on the content to teach instead of on market trends. Thus, it would be interesting having long-term robotics devices with the ability to update them partially, reducing the cost of acquiring a completely new robot, as well as a certain level of modularity to allow easy integration of new sensors, actuators, and other components, like LEGO bricks or 3D printed elements designed by students to fulfil with the proposed challenges. Furthermore, durability and reliability are also welcomed, avoiding fragile platforms that could be easily damaged.

h) *Programming languages adapted to the age:* it is recommended to program the robot with both block-based and text-based languages, allowing students to improve their skills incrementally. The selected text-based language should be a popular and highly utilized in the AI and robotics field, like Python. In this way, students will be familiar with the structure and syntaxes of widely used CI libraries, like OpenCV, Tensor Flow, or scikit-learn.

i) *Simulation models:* all the previous features apply to real robots. Nevertheless, simulation models are very important for proper robotics teaching, combining the initial stage of implementation in simulation with the validation on the real device. Using robotic simulations in classes allows the introduction of students to virtual reality topics. Furthermore, the COVID pandemic in 2019, reinforced the idea of the utilization of virtual material to continue learning when students may not share physical devices or work in groups due to health issues. However, the application of robotic simulations must support transfer to the real robot to test the students' work in real devices and face the issues that arise when moving from simulation to reality, which is still an open issue for pre-university levels [34].

Table I details the previous features in 9 of the most relevant educational robots in the market [35][36]. They are organized into three main groups. The first one addresses popular platforms in primary school and lower secondary school level, represented by the first three robots on the table. They are characterized by limited capacities for specific CI learning. The second group is composed of "not so popular" and more recent robots that could be used in secondary and high school (next three models in the table: Cozmo, Alpha AI, and Fable). They are affordable options with a high-level sensorial capacity and appropriate programming languages for CI learning. The final group (represented by the last three robots in Table I) is composed of traditional platforms, widely used in university degrees for teaching intelligent robotics, but not in pre-university education due to their high price and complexity.

In summary, there is a general lack of robotic platforms that show all the requirements for proper CI learning. In this sense, as the educational level increases, the situation improves, being a consequence of higher prices.

TABLE I

BASIC CHARACTERISTICS OF ROBOTIC PLATFORMS FOR CI TEACHING. THE COLUMN LABELLING CORRESPONDS TO THE CONVENTION USED IN THE TEXT OF SECTION II.B

Robot model	Target level	a	b	c	d	e	f (€)	g	h	i
Mbot 2 [37]	Primary	✓	×	×	×	×	150	×	✓	✓
LEGO Mind. [38]	Primary / Lower secondary	✓	×	✓	×	×	350	×	✓	×
Thymio II [39]	Primary / Lower secondary	✓	×	×	×	×	150	×	✓	✓
Anki Cozmo [40]	Secondary/ High school	✓	✓	✓	×	×	200	×	✓	×
Alpha AI [41]	Secondary/ High school	✓	✓	✓	×	×	400	×	✓	×
Fable [42]	Secondary/ High school	✓	✓	✓	✓	✓	600	✓	✓	×
Turtlebot 3 [43]	University	✓	✓	✓	✓	×	1000	✓	×	✓
Khepera 4 [44]	University	✓	✓	✓	✓	×	3200	✓	✓	✓
NAO 6 [45]	University	✓	✓	✓	✓	✓	6000	×	✓	✓

A possible solution to overcome these limitations in primary and secondary school is to use simulators, facilitating the utilization of advanced robots and their features at a low-cost. However, it must be highlighted that the most relevant robotic simulators like Webots, CoppeliaSim, or Gazebo [46] are not feasible for pre-university schools due to their high computing demand. They require installation, a dedicated Graphic Processing Unit (GPU), and a high frequency processor. In the particular case of Gazebo, a NVIDIA card is recommended, and an intel i5 with 500MB of free space [47]. For CoppeliaSim, there are no officially specific requirements, but in users' forums, the situation is similar to Gazebo. Furthermore, although robotic researchers may be familiar with the interface of these simulators, they can be too complex for secondary and high school students (and most of their features would not be necessary for those educational levels). In this sense, Open RobertaLab [48] or Robotbenchmark [49] could be interesting options for this education level as their run online and neither require installation nor advanced computational knowledge to use them. In any case, as mentioned, relying on CI education in purely simulation environments, without transfer to the real robots, is not ideal in the long-term. Hence, there is currently a gap in the educational market of robots for secondary/high school level that will be filled in the future with high probability as teachers demand them for their classes.

III. Related Work

Within the Robobo Project [10], many different challenges with a realistic setup have been proposed and tested with students in the realm of applied AI. For instance, they have faced an autonomous recycling problem [50], a collective surveillance setup [50], or an autonomous parking testbed [9]. The current research proposes the creation of a more general and complete model, where different topics of CI can be taught in the same environment at different education levels. Specifically, *smart cities and self-driving vehicles*, are becoming very popular in the educational robotics area due to the popularity of autonomous driving in the real world.

For example, authors in [51] propose a modular and integrated approach towards teaching autonomous driving. They aim to cover CI topics too, like image recognition, and others more specific to robotics, like positioning, mapping, etc. This approach is mainly conceptual, materialized on a book, and focused on the university level. Another relevant approach is the AutoAuto project [52], which utilizes the concept of self-driving cars for teaching robotics and AI to young students. The robotic platforms used in this case are toys-like cars, which utilize cameras and image recognition. The scope of the AutoAuto project is limited to STEM education under K-12. Costa et al. [53] present an autonomous driving simulator to prepare and gain the attention of the students to compete in the Portuguese National Robotic Festival (PNRT), especially in the Autonomous Driving Competition (ADC). Although the PNRT addresses both "rookie and expert" challenges, the ADC is only addressed in the "expert" challenge. Thus, this proposal focuses on students with advanced knowledge in robotics and AI, such as Gazebo and ROS.

A similar approach to the one proposed here is Duckietown [54], an online MOOC for teaching AI and robotics based on self-driving cars posted on EdX and GitHub. It is a very remarkable initiative that, in addition to instructions to construct the wheeled robots and class lessons, contains a whole application environment, with different city layouts, traffic signals, etc. This project also has its own simulation model on gym, and it utilizes both Python and ROS to program the robots. It is a project focused on high school, vocational training, and university levels, even with a research interest. The main difference with Robobo SmartCity, in educational terms, is that Duckietown is more focused on robotics than in CI, including specific lessons on mechanics and electronics in which the students have to build the robots by themselves

To sum up, self-driving cars and smart city technologies make up an application field with high potential for educational robotics and CI topics. Using such a specific application domain could seem too constrained, but the following aspects must be considered:

- It is a challenging application field, with many opportunities to face from a STEM perspective. Most of the challenges have a direct translation to real cases, increasing the motivation of students due to its real application.
- A long-term project is easy for maintenance between academic years, and it allows to achieve an outcome with real relevance and utility in CI terms.
- It promotes positive habits in students in terms of social impact of AI: reducing carbon footprint of conventional vehicles, increasing traffic safety, humanizing cities, etc.
- It proposes a more gender-neutral approach to educational robotics, moving away from classic problems such as speed competitions, football, etc. In this domain, the social aspects are key, both in the relation with people and between robots.

IV. Methodology

A. Research questions

The problem faced in this research is the *necessity of developing, in the short-term, feasible and formal teaching resources to train students in AI fundamentals*. To this end, the specific goal established has been to *design, implement, and validate with secondary and high school students, the Robobo SmartCity model*. This resource is based on the Robobo educational robot and an autonomous driving setup.

The research questions we aim to answer in this work are the following:

- Is the technology used in the model adequate to learn CI topics in secondary and high school?
    - To validate the use of robots in these educational levels.
    - To specifically validate the Robobo, the libraries, and the simulation models.
- Are the teaching units / challenges proposed within the model appropriate to learn the fundamentals of AI in secondary and high school?
    - To validate the degree of understanding of these new concepts by students.
    - To validate the practical approach to AI teaching.

- Is the proposed model a feasible resource in the long-term?
  - To validate if an autonomous driving setup covers a broad scope and it is adequate to be used throughout different educational levels.
  - To validate if the autonomous driving setup supports facing small and simple projects and also more challenging and complex ones.

B. Curriculum organization

The current section proposes a general organization of topics, skills, and platform type (real/simulated) for CI learning at different education levels, summarized in the Fig. 1. Due to different historical backgrounds, social or economic development, the educational reality worldwide is very heterogeneous, especially in primary schools. Thus, it is required to highlight that this proposal has been developed for regular education, leaving special education out of scope here. In terms of the selected topics, they have been developed analyzing the conclusions obtained from the experience gained on those countries that have already started the development of an AI curriculum for pre-university level, and which are summarized in the UNESCO report called "K-12 AI curricula" [30]. In this report, USA, Qatar, India, China, Austria, and Korea cases are analyzed, to conclude, among other aspects, that stand-alone discussion of AI topics at classes is not enough for students to properly understand them. Consequently, technical training in the basic AI topics is required at all levels, even in primary school. The specific set of topics to be included in an AI curriculum is mainly agreed in all initiatives (perception, actuation, reasoning, learning, ethics) [32]. But how to train them must be aligned with the students' cognitive development at each educational level, and also with their skills, which is more dependent on the particular educational system.

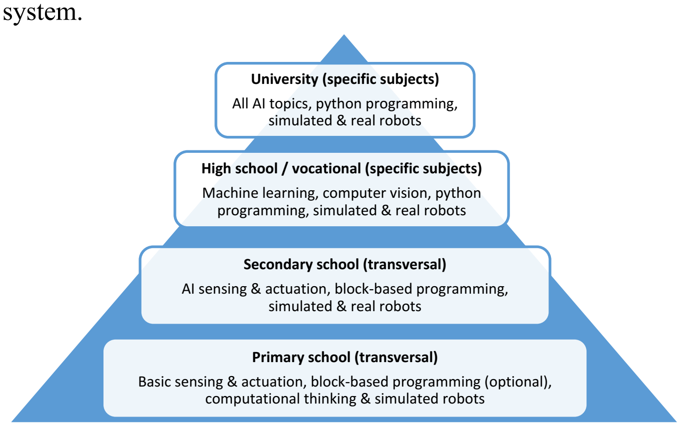

Fig. 1. Schematic organization of the main topics, programming skills and platform type for Computational Intelligence Education through robots

Hence, although it could require slight adaptations to the different national policies, the organization of topics into educational levels presented in Fig. 1 was created to respect the main conclusions from [30], and to include the authors' experience in educational robotics [9][10][11][33]. It has been structured in 4 education stages: primary school (6 to 11 years), secondary school (12 to 15 years), high school (16 and 17 years), and university degrees (>18 years). To the authors' knowledge, it is *the first attempt to provide a formal organization of topics in educational robotics for CI teaching*.

In the lower level of Fig. 1, primary school, students may be introduced to the fundamentals of sensing (distance, light, sound) and actuation (locomotion) in robotics, with simple examples that can be implemented in a robotic simulator [55]. It is very important at this age to start with computational thinking [34][56]. Thus, block-based programming could be introduced, especially in the final years of primary school. However, it is not required to use the computer to program because unplugged activities or Tangible Programming Languages (TPL) have shown to be very effective for that [57][58]. At this education stage, although specific subjects to teach CI topics have been implemented with remarkable success [14], we propose to introduce them in a transversal fashion, with small "knowledge pills" in different subjects like mathematics, science, or others.

In secondary school, students could start training in AI-specific sensing, like computer vision, sound, or tactile interaction, always with adapted levels and materials. In terms of actuation, they could be trained in the fundamentals of natural interaction, speech production, LCD screens, and, of course, locomotion. All these topics can be used in simulated and real robots through block-based programming, to provide students with a solid background in robotics before moving to specific CI learning in the next stages. Again, although these concepts could be trained in specific subjects, it is suggested to integrate them into others in an interdisciplinary way, promoting STEM learning methodologies.

Considering that at high school and vocational training levels, students who choose a technical education are more interested and motivated in science and technology, it is possible to train them in core and specific CI topics, like machine learning or computer vision. Again, using adapted resources [32][33]. A standard programming language like Python could be introduced to reach all the possibilities of its existing libraries, to focus training on the application of CI algorithms and not on developing them, which should be a matter of specialized university degrees. In this stage, due to the specialized training of the students, it makes sense to have specific subjects for CI topics, and students should reinforce the use of real robots as testing platforms, to realize the issues that arise when CI is used in the real world.

Finally, the university level is considered here to train specialized engineers, computers scientist, or similar, so all CI topics could be covered, mainly in real robots. Moreover, students here are not simple users of libraries and algorithms, but also developers. It is out of the scope of this work to detail CI education in university degrees.

With a long-term organization as the one shown in Fig. 1, it is intended that students obtain proper training in CI fundamentals, as well as basic digital skills for the 21st century that are currently presented in several fields, not only in those related to robotics and CI. In addition, students interested in AI-related degrees at university will start with a more solid background.

C. The Robobo Project

To deal with all the requirements established in section II.C for CI education, and to overcome the limitations of the robots of Table I, a smartphone-based robot is proposed here. Specifically, the Robobo robot (Fig. 2), which combines a wheeled mobile platform with a smartphone [10][59]. The platform is equipped with two DC motors with encoder on the wheels and two DC motors with encoder on the pan-tilt unit that holds the smartphone. It also has 8 infrared sensors around the base, a battery sensor, and a set of LEDs. The base communicates with the smartphone by Bluetooth. On the other hand, the smartphone provides Robobo with state-of-the-art sensors and actuators, as well as high computing and communication capacities. Most current smartphones contain various sensors, like two high-resolution cameras, a microphone, a 3D gyroscope, a 3D accelerometer, a 3D magnetometer, a tactile screen, a light sensor, and GPS, among others. Regarding actuators, the smartphones have an LCD screen, speakers, and flashlights. While in terms of connectivity they are endowed with Bluetooth, Wi-Fi, 4G or NFC among others.

Fig. 2. The Robobo robot used in the Robobo SmartCity model

The Robobo Project [60] is based on two more elements: a collection of teaching units, and a programming framework. Both have been organized in three levels, with specific resources adapted according to the student's age and skills. (1) A starting level, for those who give their first steps in robotics. This level is based on Scratch3 [61] as a programming language, and the RoboboSim [62] simulator; (2) An intermediate level, focused on those students who already have a robotics and programming background. Python is the selected programming language for this level, and RoboboSim or CoppeliaSim as recommended simulation models. Gazebo could also be used with Python, but it is not recommended for this educational level; (3) Finally, the advanced level allows users to utilize ROS [63] framework to develop their programs, which can integrate all the functionalities available on it. For this level, the recommended simulator is Gazebo. Fig. 3 shows a diagram containing the basic elements required to program Robobo. The programming resources and simulation models are available at the Robobo wiki [64], where it is possible to find the different elements of the model, both in the table of contents and on the wiki pages.

All the programming levels have access to the same set of modules organized in libraries that cover different areas of the robot capabilities [50]:

- *Computer vision*: face detection, tag recognition, color tracking, object identification, QR detection, and lane recognition are available. In addition, video streaming is provided to run external libraries over the image (real and simulated).
- *Sound processing*: musical note detection and production, clap detection, and noise level.
- *Remote Control*: WebSocket remote interface, ROS remote interface.
- *Speech*: speech production and recognition.
- *Emotions*: facial expression and mood sounds.
- *Touch*: touch screen tactile gestures detection.
- *Other sensing capacities*: smartphone sensing capabilities like accelerometer, gyroscope, and light level (depending on the characteristics of the smartphone).
- *Rob Interface*: access to the robotic base sensors and actuators, namely, wheel motors, smartphone holder motors, motor encoders, IR distance sensors, battery level, and LEDs.

D. City layout and traffic signs

The smart city proposed in this resource is a scaled model of a city neighborhood (Fig. 4) represented by a rectangular city layout of 3.5m x 4m. The layout is made up of an external two-way road, surrounding a central part that contains a roundabout, where four two-way road sections intersect.

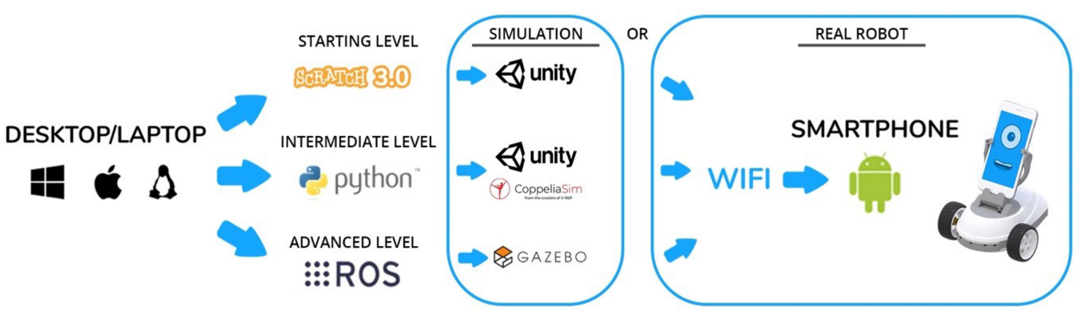

Fig. 3. Robobo programming options considering simulated and real robot

The city layout is organized into five different sectors (Fig. 4 bottom), occupied by different buildings and city facilities, created to represent a realistic people flow in a city environment. It is important to mention here that, to avoid illumination problems, the buildings included in sectors A and B should not be too high in the real setup. At sector D we have included a parking area (see top image of Fig. 4). In it, there are four different parking places, allowing parking in 2 different ways:

- *In-line parking*. Delimited both by road marks and vertical signs.
- *Perpendicular parking*. Delimited only by road marks and by a “P” letter.

The parking area also represents the charging area for the robots, similarly to real charging stations for electrical vehicles.

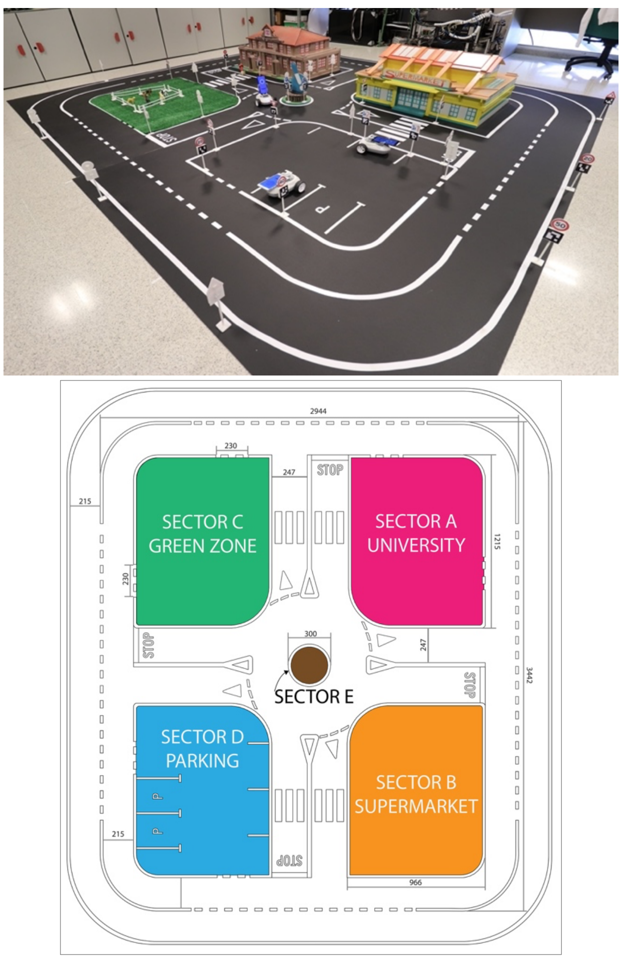

Fig. 4. Top: Pictures of the real mockup of the city model. Bottom: Layout map of the mockup

Traffic signs are a key element for traffic regulation. We have created vertical and horizontal signs at Robobo SmartCity:

Vertical signs. They are 0.23 m in height, with a square base of 0.035 m of side (Fig. 5). Each signal has different sign plates accordingly to the design of the traffic signs of the real world. In addition, they also have an auxiliary plate with velcro, to include a tag that simplifies the sign identification with computer vision. They have been created with ArUco markers and QR codes.

Four different types of vertical signs were implemented, which make up a total of 13:

- Mandatory signals. It was decided to design only two mandatory signs, right turn, and left turn direction.
- Prohibition signals. The six prohibition signs designed are four speed limit signs (10, 20, 40, and 50), together with the STOP and yield signs.
- Indication signals. The roundabout and parking signs have been included.
- Warning signs. Three of these signs have been designed: dangerous right turn, dangerous left turn, and pedestrian crossing.

Fig. 5. Vertical traffic signs designed for the model

The 3D models of these signs can be downloaded from the Robobo GitHub repository and printed at schools.

Horizontal signs. Horizontal signs are painted on the roadway. They clarify limits or areas of influence of the signage, such as the zebra crossing or the height at which a stop must be made. Although these signs could be detected using the camera, they were mainly included for realism.

It should be pointed out that this city layout is a proposal, and it can be modified and adjusted by the teachers accordingly to the student level. For example, in section V, a project carried out by secondary school students who utilized a simplified version of the full layout will be presented.

E. Simulation models

The Robobo SmartCity was originally designed to be used in the real world, with the real Robobo platform. Although we recommend doing it and engaging students in the whole process, constructing the real mockup could be complicated, expensive, or consuming too time for teachers and students in many schools, so we have created different simulation models of it. They are completely free and accessible from the Robobo wiki. Specifically, the three models are:

- *RoboboSim.* This model corresponds to a specific Robobo simulator created using the Unity technology. It creates a realistic, simple, and computationally light 3D simulation. As Unity is used for developing video games, RoboboSim has usability and aesthetics like them, making it familiar for young students, helping to increase its acceptance. Thus, it is the simulation tool recommended for the first levels of learning, although it has also been used at the graduate level. It runs under macOS, Windows, and Linux, and it is compatible with the Robobo blocks library for Scratch3 and the Robobo Python library. Consequently, students can take advantage of using basic AI functionalities at this age, like color recognition, QR detection, speech production, emotion production, or tactile interaction. Switching from simulation to the real robot is straightforward, simply changing the IP address field, as it is shown in the wiki of the project.

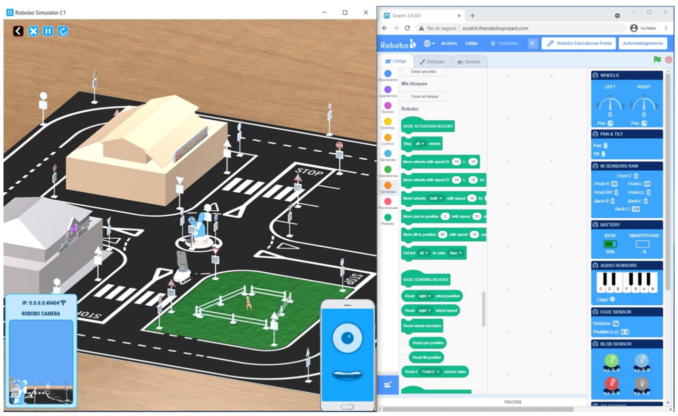

Fig. 6. Left: RoboboSim model of the Robobo SmartCity. Right: Scratch 3 IDE with the Robobo blocks

The left image of Fig. 6 shows a snapshot of the RoboboSim model of the smart city. All the elements described above are presented in the figure, although some of them have been simplified to reduce computational costs. The right image corresponds to the Scratch 3 interface with the Robobo specific blocks loaded and with the monitoring window on the right.

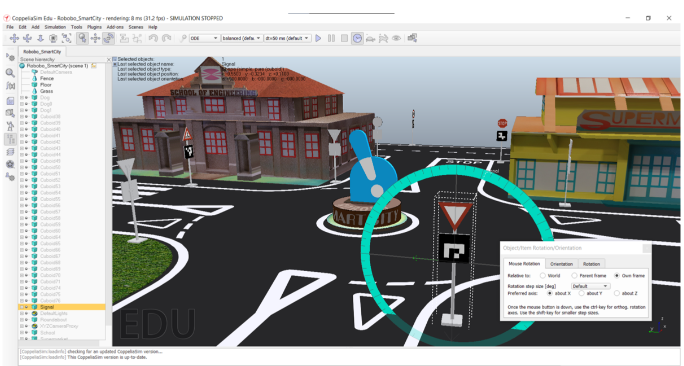

Fig. 7. CoppeliaSim model of the Robobo SmartCity

- *CoppeliaSim.* It is a 3D model that runs under the CoppeliaSim simulator. It is a more powerful simulator than RoboboSim, especially in terms of the possibilities to control the scene, the physics, and the dynamics of the simulation objects. For example, it is possible to add new objects on the scene, modify their properties, both visual and dynamic ones, as well as create independent behaviors over them, among other features. To support all these features, the CoppeliaSim needs installation. Furthermore, the simulation can be very demanding in terms of computer resources, being this characteristic its main drawback for schools. A predefined Robobo SmartCity layout has been included in the wiki (Fig. 7), with the possibility of being completely modified and adjusted to the teacher's and student's requirements. It also allows the modification of the Robobo simulation model.

  This simulation model is suitable for intermediate and advanced students who pursued more digital skills in terms of 3D design and programming. Hence, only Python language is supported. The bridge for using the CoppeliaSim model and the real Robobo is straightforward, and it is documented on the Robobo wiki. Furthermore, this model allows the use of functionalities from other advanced Python libraries, such as OpenCV and Tensorflow, giving an idea of the potential of this simulation model.

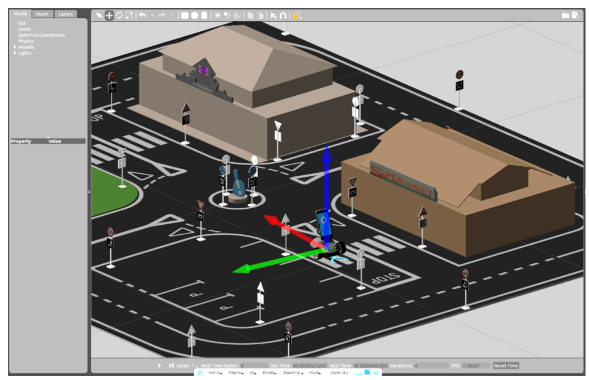

Fig. 8. GazeboSim model of the Robobo SmartCity

- *GazeboSim.* The last simulation model runs under the Gazebo simulator. It focuses on university students or researchers in intelligent robotics that use ROS1. The model shares similar functionalities with the CoppeliaSim one, such as allowing the complete modification and adjustment of the city layout and the Robobo model. It is compatible with Gazebo 11 (ROS Noetic), and it can be programmed using the Python library. Again, it is possible to include different external libraries for machine learning, computer vision, etc. The transfer between simulation and the real robot is quite simple as it is described in the wiki. Fig. 8 contains a snapshot of this simulation model.

F. *Specific libraries*

A series of libraries with different functionalities have been developed to help and guide students, mainly in initial and intermediate levels, for performing realistic programs in the city. These libraries are:

- *Lane detection.* This library allows Robobo to follow the lanes of the city model, both the continuous and discontinuous ones (see Fig. 9). It is a basic feature in autonomous driving vehicles to remain in the lanes that delimit the roads. It is also suitable for curved lanes with a high degree of curvatures, such as the roundabout or 90° curves. Its implementation is based on the Canny algorithm for edge detection and uses the Hough transformation line detection. In addition, it utilizes features from the OpenCV library for image filtering and noise elimination [65]. From the output provided by this library, the slope and offset of a line, students could implement their specific Robobo controller.

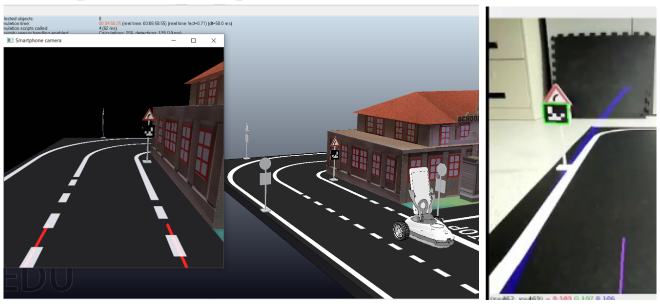

Fig. 9. Lane detection in the simulated (left/middle) and real (right) models

- *Object recognition*. It allows Robobo to identify different objects that may encounter during navigation, as displayed in Fig. 10. The object recognition system employs a preconfigured neural network based on Mobile Net as a recognition algorithm [65]. Although it is possible to download it already pre-trained with a series of objects, it was trained from scratch using the machine learning framework Tensorflow with a set of specific objects relevant for this model [65]. Task that can also be performed by students and help to teach them how an Artificial Neural Network (ANN) works.

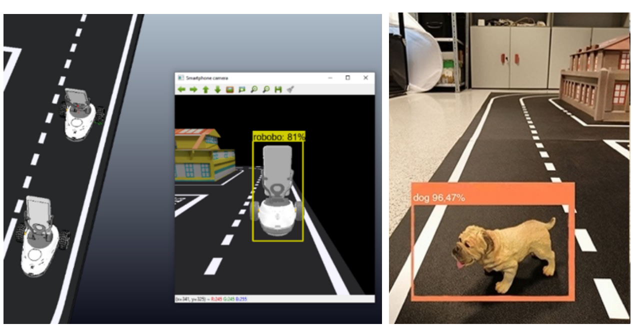

Fig. 10. Object recognition library detecting other Robobo (left) and a dog doll (right)

- *ArUco detection*. This library allows ArUco fiducial markers detection, which can be used as artificial landmarks for robot location or, in this case, to identify traffic signs [65]. For the correct utilization of the library, the smartphone camera must be calibrated before using it. To do so, we have developed an Android app to simplify this process (right image of Fig. 11), which returns a camera matrix and a distortion vector, required to estimate the pose of the ArUco. With this library, students can detect many traffic signs with high accuracy, due to the adequate properties of this type of tag for visual detection (left and middle images of Fig. 11).
- *Traffic sign recognition.* It allows detecting the vertical traffic signs using the camera, without relying upon ArUco or QR tags (see Fig. 12). This library is based on a multilayer perceptron ANN and a dataset of real traffic signs. It performs some simple pre-processing stages over the image using OpenCV like a greyscale conversion and a HOG method before introducing the image in the ANN, in the same way as real car systems do. Hence, it is interesting to explain it to students, who can understand how such real car systems work without requiring a tag.

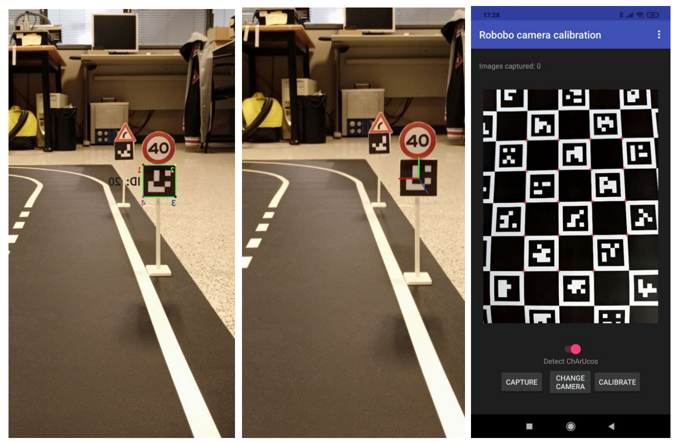

Fig. 11. Left and middle: Aruco markers detected by the library. Right: calibration app

These four libraries are only available in the Robobo Python library, so they cannot be used with Scratch3. They can run remotely on a computer using the video streaming library, and the first three can also run natively on the Robobo's smartphone, being this a very exclusive feature of this model. In the case of Scratch3, students can detect traffic signs using QR codes, which is an already implemented programming block. Of course, other Robobo libraries can be used in this model, like IR sensing, motor commands, speech production, and others.

Fig. 12. Three examples of successful traffic sign identification with the library

Furthermore, although these libraries have been developed in Python particularly for Robobo, they are not exclusive to it.

they are accessible worldwide in the GitHub repository and can be downloaded, modified, and deployed in any other robot, scenario, or environment, allowing teachers and trainers modify them as they desire.

G. Challenges in the Model

This section contains a set of possible challenges that can be accomplished in the Robobo SmartCity at different educational levels, taking advantage of the city design and the functionalities developed both for the real implementation and the simulation ones (Table II). These challenges have been organized by educational levels according to the Fig. 1. Furthermore, each challenge shows a set of incremental tasks towards a final objective of including them in a global task, if teachers want to use it in the long-term to train students in specific CI topics. Table II also contains challenges for University level, as an example of more advanced topics that we consider should be out of the scope in pre-university level, and only relevant for technical degrees.

TABLE II

EXAMPLES OF POSSIBLE CHALLENGES TO BE PROPOSED TO STUDENTS ACCORDING TO THEIR EDUCATIONAL LEVEL

Challenges
SECONDARY SCHOOL (RoboboSim, real robot)
Move autonomously from one fixed point to a target location, respecting traffic signs and avoiding collisions (there are no other robots in the city)
o Understand the concept of autonomous control, programming the robot without relying to specific values or thresholds but on sensing
o Detect objects using IR sensors and avoid collisions (any object that appears suddenly on the road)
o Detect traffic signs using QR codes and respond to them (adapting speed, stopping appropriately)
o Detect colored objects that could appear on the road
o Straight movement without using lane detection, based on the motor encoder information and orientation sensor (kinematics)
o Speech and emotion production informing the robot status (natural interaction)
Autonomous parking
o Perform reliable parallel parking on free spaces using QR codes, IR sensors, encoders, and orientation sensors
o Coloured landmarks could be introduced in this area if the colour detection blocks aim to be used
Reality gap
o Introducing the reality gap problem: understanding the differences between simulation and the real world, the simulation is a simplified version of the reality
o Virtual versus augmented reality
HIGH SCHOOL (RoboboSim, CoppeliaSim, real robot)
Move safely and autonomously following lanes from one fixed point to another location respecting traffic signs and avoiding collisions. Other Robobo could be in the city, so navigation must be adapted to the traffic and basic communications are allowed
o Detect traffic signs using ArUco tags and respond to them (adapting speed, stopping appropriately)
o Detect objects using the vision library and avoid collisions (person-like dolls, pet-like dolls)
o Detect other Robobos using the vision library, and adapt robot behaviour to them
o Detect objects using IR sensors and avoid collisions (any object that appears suddenly on the road)
o Speech and emotion production informing the robot status
o Basic communication between Robobos, to exchange traffic information. For instance, if a road section is closed
o Straight lane following control for fixed speed (curves could be detected using traffic signs and QR/ArUco tags)
o Energy consumption control, moving to the parking area if required
o Machine learning fundamentals: supervised learning of an image classification model of simple elements that could appear in the road, reinforcement learning to perform the simple task of moving straight without colliding with other Robobo or similar
Autonomous parking
o Parking area identification using the corresponding traffic sign
o Detect free spaces utilizing of the QR codes
o Program reliable serial of parallel parking on free spaces using QR codes, IR sensors, encoders, and orientation sensors
o Introduce recharging zones marked with colours, so robots with low energy must use them
Reality gap
o Identify phenomena of the real world, like sliding or clearance, for example, that may cause unexpected behaviors but that are usually omitted in simulation
o Understanding the current concept of digital twin and why it is useful
UNIVERSITY (CoppeliaSim, GazeboSim, Real robot)
Navigate optimally and safely to a destination following the lanes, respecting traffic signs, and avoiding collisions. Traffic information must be interchanged between Robobos, which can adapt their path planning
o Map creation and storage (graph theory)
o Global positioning system (only in simulation)
o ArUco marker localization
o Path planning to find optimal route (graph search)
o Optimal navigation using IR sensors, orientation, and encoders
o Basic information exchange between robots
Exchange advanced information between Robobos and with a central traffic control center, so traffic signs could be only complementary for vehicles, and the overall response of the system is optimal in terms of congestion avoidance and energy consumption
o Advanced information exchange between robots, including ontology definition
Driving autonomously in the roundabout
o Detect the direction of driving, the size of the roundabout, and adapt to the driving speed
Computer vision development, so current libraries are improved
o Development of an improved object recognition model, with new objects to be included by students
o Development of an improved traffic sign identification model, training it with pictures taken by the students and avoiding the use of tags
Autonomous behavior
o Design a reactive architecture to perform basic navigation
o Develop a reinforcement learning model to control navigation

H. Data collection instruments

To answer the research questions proposed above, online questionnaires were filled by students during the educational intervention sessions. We relied on the Microsoft Forms application used through a web browser in a completely anonymous fashion.

I. Data analysis procedures

Collected data was analyzed using a spreadsheet software applying basic statistical procedures. The main measure used in the analysis was simply the percentage of students per answer.

In addition, a specific measure was defined as the *Group Success Rate* (G_{SR}), an objective value that is computed from all the students' answers to a given question as:

$$G_{SR} = \frac{number\ of\ correct\ answers}{total\ number\ of\ answers} * 100$$

Hence, this is a percentage of the students that provided a correct answer to the question. It must be pointed out that some questions have multiple answers, which has been also considered in this measure.

V. Results

The following two sub-sections are devoted to presenting the application results of the Robobo SmartCity model with secondary and high school students. The goal is to answer the research questions established in section IV.A. The first sub-section contains validation results obtained from workshops developed in the realm of the AI+ project [33] in 2021 and 2022. With them, the two first research questions are faced: (1) Is the technology used in the model adequate to learn CI topics in secondary and high school? (2) Are the teaching units / challenges proposed within the model appropriate to learn the fundamentals of AI in secondary and high school? The second section contains a more detailed activity, carried out with two small groups of students but with a more challenging goal in technical terms. With it, the third research question is answered: (3) Is the proposed model a feasible resource in the long-term?

A. CI workshops with secondary school students

The AI+ project was an educational project funded by the European Commission between 2019 and 2022 within the Erasmus+ programme, which encompassed 7 partners of 5 different European countries with the aim of developing a curriculum about AI for pre-university students [33]. The authors of the current paper were part of the leading team of this project at the University of Coruña, who developed the Teaching Units (TUs) that make up the curriculum. The TUs are the final results of the project and available online [33]. The other 6 partners were secondary schools, including teachers who revised and tested the TUs with their students, and provided feedback to the leading team to improve them.

The model presented here was used in different workshops to test the TUs focused on intelligent robotics. In all cases, the students did not have any previous training on the specific topics of the workshop. This was a pre-requisite of the AI+ project for the students' selection. One of these workshops was carried out in blended fashion in May 2021 at the University of Coruña. A total of 41 students participated in the workshop, 24 from Spain, 6 from Italy, 6 from Slovenia and 5 from Lithuania. From these 41 students, 31 were male, 8 female, and 1 preferred not to give that information. The age range was from 15 to 19 years old, being the students between 15 and 16 years old most of them (73.17%) The workshop lasted for 6 hours (2 sessions of 3 hours each). 29 of them attended the workshop in person, while 12 attended online due to the COVID pandemic. The RoboboSim simulator was used to face the challenge proposed in the TU, programming it in Scratch3. Once solved, students validated their solutions in the real Robobo robot. The materials used in this workshop are available at https://bit.ly/3BYke57.

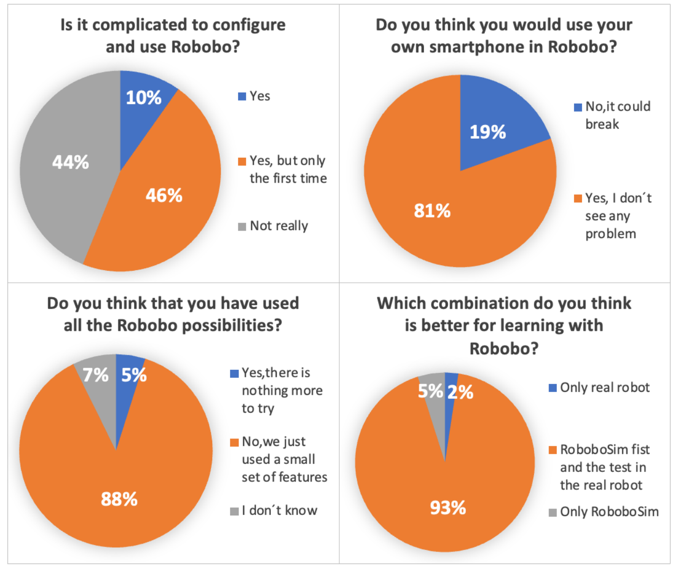

Fig. 13. Answers to the questionnaire about the model adequacy

The goal of this workshop was twofold. On the one hand, to obtain direct feedback from students about the usability and adequacy of the model for CI learning, mainly the Robobo robot and the RoboboSim. To this end, a questionnaire was filled by students at the end of the second session. Fig. 13 contains their answers. It can be observed how the model seems to be technically appropriate for this scope, easy to use, and the combination of simulation and real robot is clearly the best accepted by them.

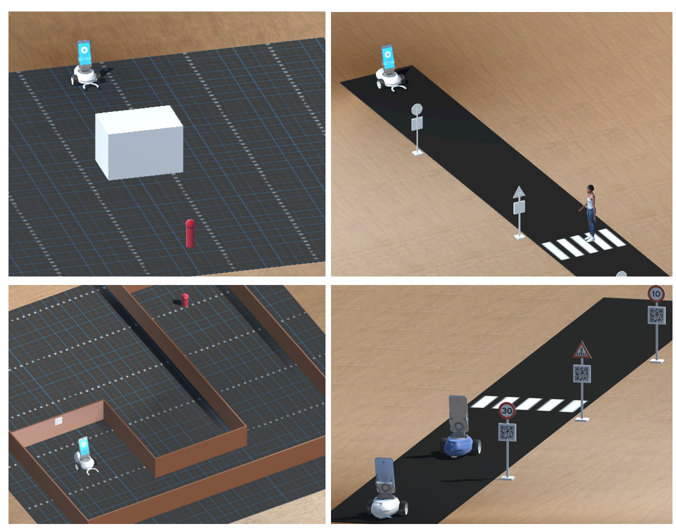
Fig. 14. Snapshots of the RoboboSim worlds used in workshop 1 (top) and in workshop 2 (bottom)

On the other hand, regarding CI topics, the workshop aimed to develop the TU7 of the AI+ curriculum, focused on introducing autonomous navigation with robots, and evaluate the learning outcomes. Specifically, the challenge in this TU was to make the robot move straight in a path without colliding with an object/pedestrian placed at random positions, using for that IR sensors, orientation sensors, wheel motors and appropriate programming routines (see top images of Fig. 14). According to the first two points of Table II, this challenge provides students with the fundamentals of what autonomous navigation means, and what it is necessary to achieve it in terms of sensors, actuators, and control. Moreover, the reality gap, commented also in Table II for secondary school, was introduced, and students had to slightly adapt the code from RoboboSim to make it work properly on the real robot. The main update was in terms of the thresholds used in the IR and orientation sensors.

Another questionnaire was filled to evaluate the proposed model at the end of the 2nd session. For the sake of clarity, students' answers to just three of these questions are shown in Table III, while the others can be accessed at https://bit.ly/3BYke57. Correct answers are displayed in green color. As it can be observed, learning results were successful for most of the group members.

A second workshop was held in Lithuania in February 2022. It was carried out in blended fashion too, but in this case, only 24 students attended, 18 of them in person. The students were from Spain, 12, Slovenia, 6, Lithuania 5 and Italy 1, being 20 male and 4 female and with ages in the range of 15 and 18 years old. The goal was to implement TU11 and TU12 of the AI+ curriculum, each of them in a 4-hour session. The RoboboSim was used as the main tool (bottom images of Fig. 14 show two of the proposed worlds), although the programs were also tested in the real robot at the end of each session. Phyton was the programming language used to face the challenges.

TABLE III
GROUP ANSWERS TO THE QUESTIONNAIRES ABOUT AUTONOMOUS ROBOTICS (WORKSHOP1)

Are all robots autonomous?		Are simple sensors like distance or orientation useful for intelligent robotics?		What should we avoid when programming an autonomous behavior in a robot?	
Yes, all of them are autonomous	2,4%	No, only for simple robotics	7,3%	Using sensors	9,8%
No, no robot is completely autonomous	17,1%	Yes, because they are required to solve low-level tasks	80,5%	Collapsing the program with infinite loops	36,6%
No, some are and some are not autonomous	80,5%	They could not be, the important is the camera	12,2%	Using pre-defined times or distances	53,7%

TABLE IV
GROUP ANSWERS TO THE QUESTIONNAIRES ABOUT COMPUTER VISION (WORKSHOP2)

The X and Y coordinates of a color blob correspond to…		Which of the following features makes computer vision hard in real robots? (it can be more than one)		A color blob is..	
Any point in the blob	4,2%	The speed of the robot	29,2%	A region of an image with a homogeneous colour value	20,8%
The coordinates of the center of the blob area	75%	The illumination conditions	75%	A group of pixels of the same colour	29,2%
The coordinates of the border of the blob area	20,8%	The speed of the computer	25%	Both are correct	50%
		None of them	16,7%		

TU11 was focused on computer vision basics. Students faced the challenge of finding a colored object at an unknown position in the environment and moving towards it using the information obtained with the smartphone's camera. To solve this challenge, they learned the fundamentals of image segmentation, color blobs, and how to reference blob positions on an image. This problem is included on Table II for secondary school level, as part of the autonomous navigation and autonomous parking challenges. At the session completion, students filled a questionnaire with different technical questions about computer vision. Three of them are shown in Table IV while the others can be accessed at https://bit.ly/3y59jW9. Most of the group achieved a proper understanding of the trained concepts, according to their answers. The question in the third column of Table IV was tricky for students, because both answers were correct, so the achieved result is not bad.

TABLE V
GROUP ANSWERS TO THE QUESTIONNAIRES ABOUT REINFORCEMENT LEARNING (WORKSHOP2)

Reinforcement learning is about...		Q-learning tries to…		A Q-value indicates…	
Learning through rewards and penalties	87,5%	Minimize errors	29,2%	If the robot will crash	20,8%
Learning in a strong way	12,5%	Maximize reward	41,7%	The new reward	16,7%
No idea	0%	Learn the reward	29,2%	How good an action is	62,5%

In the second session, students implemented TU12, focused on the fundamentals of reinforcement learning (RL), a key topic in CI, mainly in autonomous driving. This topic has been included in Table II for high school level, but it was tested here with the special case of the secondary school students involved in the project. The goal to be achieved in this workshop was to develop a Q-learning algorithm so the robot autonomously learns to move safely over a path without colliding with objects nor getting out of the trail.

Again, at the end of this second session, students filled a questionnaire with different questions about RL. In this case, the questions were more technical, and results worsened as shown in the three example question of Table V (additional questions are at https://bit.ly/3y59jW9). This does not mean that the proposed model is not valid for complex topics, but more sessions and time are required. This will be clearer with the result explained in section V.B.

To provide a more general measure of the model reliability in terms of CI topics learning, all the questions included in the 3 questionnaires of workshop 1 and workshop 2 were analyzed in terms of the *Group Success Rate* (G_{SR}) defined in section IV.I.

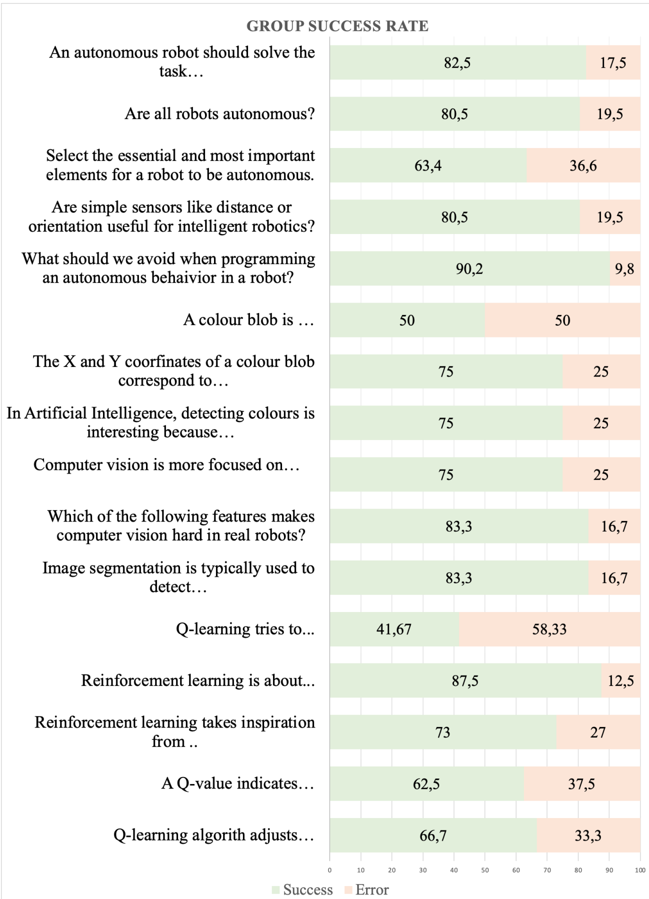

Fig. 15. Group success rate for the questionnaires in green color

Fig. 15 displays a representation of the G_{SR} obtained in the workshops (in green). Overall, the students answered the questions correctly, with the exception of those two previously commented ($G_{SR} < 50$). This measure must be analyzed considering that students did not have any previous training on the topics and, more important, they learned about them in a fully practical fashion. Consequently, G_{SR} could be taken as a measure of the students' improvement in one session based on the proposed model.

B. STEMBach project at high school level

The STEMBach program [66] is an educational initiative from the Galician region government in Spain that aims to introduce high school students into scientific research before they access university. They must develop a scientific or technical project for one year, write a paper (adapted to their educational level), and carry out a public presentation in front of a tribunal. These students have an adequate technical background, and they are highly motivated and proactive in STEM projects. The final report created for these groups are available at https://bit.ly/3SDqA0Q.

In the Robobo SmartCity, two groups of two students, each one from a different school (IES Concepción Arenal and IES Sofía Casanova) at Ferrol region (Spain) carried out two similar projects. The main research objective of the projects was to *analyze from a practical perspective the benefits of autonomous driving in traffic safety*. One of the projects was carried out in a simplified version of the real mockup and programmed in Scratch3 (group A), while the second one was realized in the complete mockup described above and programmed in Python (group B). The work was organized in sessions of 4h per week, two at the school and two at home, due to the COVID pandemic. To reach the final goal, the students' work was organized following the same methodology:

1) Develop a control program to maintain the straight movement of the robot

Using the encoder sensor of the wheel motors, students developed a program that allowed the robot to move straight, avoiding the typical imbalances between the two motors of this type of robot. This challenge allows us to introduce them to the concept of proportional control in a simple way. In addition, they understood the fundamentals of odometry, by working with encoder sensors and realizing that the motor command could be different from the motion finally achieved in the real robot. In addition, this exercise also showed the relevance of programming based on the robot sensors instead of fixed constants, because the robot movement needs to be constantly corrected, but with a different magnitude each time.

2) Develop a sensing program to detect traffic signs using the smartphone's camera

Students used the available libraries to detect a reduced set of traffic signs. Group A used QR codes in the traffic signs, while group B used ArUco tags. The objective of their program was to identify the traffic sign and the distance to it with reliability, providing a proper response in the next step. Students had to understand the basics of computer vision, how it is influenced by environmental conditions, and how the view angle of the robot must be carefully adjusted.

Fig. 16. Left: The "pedestrian" detected by secondary school students. Right: The pedestrian-like doll used for high-school students

3) Develop a control program to adapt Robobo's response according to the traffic signs

Each of the traffic signs required an adequate response from the robot. These responses were programmed by the students. A previous experimentation process was required in some cases to empirically adjust the robot control. For instance, in the case of the STOP signal, with the collected data, the students performed a simple regression analysis, and a linear model was obtained to control de robot speed depending on the distance to the signal, to avoid sudden stops. Details about this work are provided at https://bit.ly/3SDqA0Q.

4) Develop a sensing program to detect pedestrian miniatures in the pedestrian walkways using the smartphone's camera and modify the control accordingly

This program implied detecting a miniature model of a pedestrian near a walkway with the camera and acting accordingly with the robot.

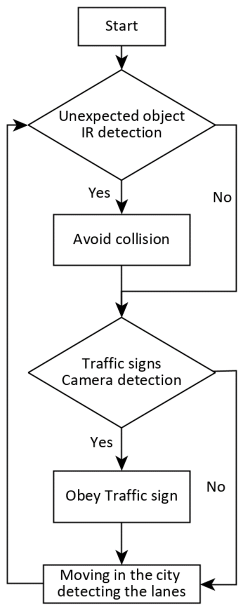

Fig. 17. Flow diagram of the control architecture created by high school students

For group A, this detection was carried out using the color detection block of Scratch3, and the pedestrian model was simply a colored ball (left image of Fig. 16). In Python, the detection was performed using the object detection model explained in the previous section, which provided the *person class* together with its location data (right image of Fig. 16). To properly implement this control, students had to understand the basics of color and object detection with a camera and consider again illumination and view angle issues. Moreover, they had to understand the details of these methods, and how cartesian coordinates on the robot view can be transformed to the 3D world, to calculate when the pedestrian crossed the street completely.

5) Develop a sensing program to avoid collisions with unexpected objects, using the IR sensors

The IR sensors of Robobo were used to avoid collisions with objects that suddenly appeared on the view of the robot. Regarding this task, during the evaluation of the student's presentation, one of the evaluators mentioned a relevant point that was not addressed either by the student or by their supervisor: this should be the priority task of Robobo, rather than traffic sign detection or lane detection because the fast detection of IR sensors helps to avoid both collisions with objects or other robots, but also pedestrian run over, given de maximum priority to the people security, as it happens in the real world.

6) Develop a simple control architecture that integrates the previous control and sensing programs in coherent way, so priorities in terms of traffic safety are respected

The global program of students had to control different aspects in real-time, so it was required that they developed a logic of priorities in it. Such control architecture was the same in both groups, although high school students used lane detection. The flow diagram they developed for their architecture is displayed in Fig. 17. They had to understand something that it is not intuitive for the students: following the lanes is the block with the lowest priority, although it is the most used. Only if no unexpected object is detected and no traffic sign is detected, the lanes must be respected.

7) Test the control architecture in the real mockup, with different traffic sign positions, pedestrian's presence, and unexpected objects.

The global program was tested by students in the real mockup. To validate the reliability of the solution, the following variations on the environment were introduced:

- A different sense of movement of Robobo (to find left and right curves).
- Different traffic sign positions.
- Presence and not the presence of pedestrians on the walkways.
- Presence of unexpected objects on the road.

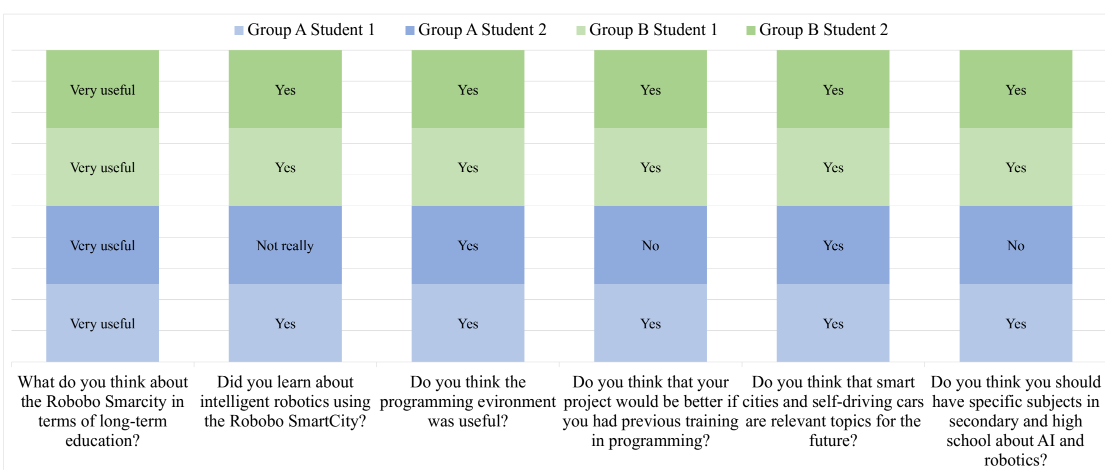

Fig. 18. Results of the survey filled by STEMBach students after finishing their project with the model

The final objective of this project was to achieve an autonomous response on the robot, moving permanently in the different setups, according to the traffic signs. It was properly achieved in both cases, obtaining the highest qualification.

At the end of the project, the four students filled a short survey after finishing their project, to provide us with anonymous feedback about the model. The answers to some of the questions are displayed in Fig. 18 Although the number of participants is too low to obtain statistically relevant conclusions, there are interesting aspects to consider. The first two questions reinforce the utility of the Robobo SmartCity model in terms of long-term potentiality and relevance for the students. Middle ones, related to the Robobo programming options, point out the necessity of providing students with previous programming training and environments that cover different educative levels. Finally, the two right questions show how most students were able to understand the relevance of intelligent robotics in their future.

VI. Discussion and Conclusions

Throughout this paper, an educational resource for training students in Computational Intelligence (CI) topics has been presented. The Robobo SmartCity model was designed and motivated by the relevance that current digital education plans are giving to advanced computer science topics, such as AI and autonomous robotics, at all education levels. Thus, formal resources are needed to help teachers at different levels to teach these new topics to students in a feasible way.

As a starting point, this work proposes a tentative organization and formalization of CI topics and skills to be trained at different education levels. To that, a set of requirements for the robotic platforms to match the requirements of this organization has been provided, putting strong emphasis on being accessible and affordable by education institutions, specially, in the long-term.

Simulation models, programming libraries, a city layout environment, and the real robot Robobo have been included in the Robobo SmartCity teaching resource, to train students in CI topics, like autonomous control, computer vision or machine learning. The resources are open source, leaving the possibility for teachers and students not only of sharing their experiences with others all around the world, but also the possibility of modifying it and customize the resources for their projects.

Three research questions have been proposed in this study. The first one, focused on testing the validity of the technology. The second addresses the adequacy of the teaching units and challenges, and the final one, evaluating the long-term possibilities of the model.

To answer the research questions, the educational model has been tested in secondary school, high school, and university degrees [67] during years 2019 and 2020, although this paper only addresses secondary school and high school. The validation was performed in two different workshops within the AI+ project, and also as a realm of the STEMBach project. In the first case, students from 6 different countries with no previous training in CI topics used the model to learn about autonomous navigation, computer vision and reinforcement learning. They showed a remarkable level of knowledge acquisition on the topics addressed in the workshop in a short period of time, as can be extracted from the results of Fig. 15, based on the Group Success Rate.

In the second test case, two groups of students were involved in a higher-level educational project with the aim of improving the traffic safety increasing the robot autonomy. The results showed the model learning possibilities in long-term goals, as well as an example of how to develop a scaffolding learning methodology along with an educational level.

Globally, it has been shown that the degree of improvement in the educational objectives was very high, as well as the acceptance level of the proposed model. On the one hand, students acquired a technical background in the covered topics in a short period of time, setting the basics for a prior knowledge in CI and allowing them to continue their technical background with confidence, due to this initial knowledge acquired. For those students who will not be interested in technical degrees, these workshops have helped to give them a basic impression of how autonomous robotics operates in real environment and the problematic they must face on. On the other hand, some concepts were not well understood by students, showing that although Robobo SmartCity it is suitable tool for teaching CI topics, specific workload is needed for complex topics. For example, special emphasis can be made by the teachers in those topics that could be hard to learn by students, by presenting different examples or scenarios where to clearly show those differences.

As commented in section I, the main goal of this paper is to foster the utilization of the presented model as a starting point for advancing in the formalization of CI training, mainly at pre-university education. Teachers and educators are encouraged to download and try this teaching resource and provide feedback and improvements for the whole community.

Acknowledgment

This work has been partially funded by the Erasmus+ Programme of the European Union through grant number 2019-1-ES01-KA201-065742, and the Centro de Investigación de Galicia "CITIC", funded by Xunta de Galicia and the European Union (European Regional Development Fund- Galicia 2014-2020 Program), by grant ED431G 2019/01. In addition, the “Programa de ayudas a la etapa predoctoral” from Xunta de Galicia (Consellería de Cultura, Educación y Universidad) supported this work through Sara Guerreiro’s grant.

References

[1] Lane, M. and A. Saint-Martin, “The impact of Artificial Intelligence on the labour market: What do we know so far?”, *OECD Social, Employment and Migration Working Papers*, No. 256, OECD Publishing, Paris, 2021, https://doi.org/10.1787/7c895724-en.

[2] Szczepański, M. (2019). Economic impacts of Artificial Intelligence. *European Parliamentary Research Service.*

http://www.europarl.europa.eu/RegData/etudes/BRIE/2019/637967/EPRS_BRI(2019)637967_EN.pdf. Accessed December 2022.

[3] Schiff, D. (2021). Education for AI, not AI for Education: The Role of Education and Ethics in National AI Policy Strategies. Int. J. of Artificial Intelligence in Education. https://doi.org/10.1007/s40593-021-00270-2.

[4] Miao, F., & Holmes, W., "AI and education: Guidance for policy-makers". UNESCO, 2021, https://unesdoc.unesco.org/ark:/48223/pf0000376709.

[5] Zimmerman, M., "Teaching AI: Exploring new frontiers for learning". Portland, OR: *International Society for Technology in Education*, 2018.

[6] J. M. Esposito, "The State of Robotics Education: Proposed Goals for Positively Transforming Robotics Education at Postsecondary Institutions," in IEEE Robotics & Automation Magazine, vol. 24, no. 3, pp. 157-164, Sept. 2017, doi: 10.1109/MRA.2016.2636375.

[7] J. E. Bezerra Junior, P. G. G. Queiroz and R. W. de Lima, "A study of the publications of educational robotics: A Systematic Review of Literature", in *IEEE Latin America Transactions*, vol. 16, no. 4, pp. 1193-1199, April 2018, doi: 10.1109/TLA.2018.8362156.

[8] Tzagkaraki E., Papadakis S., Kalogiannakis M., "Exploring the Use of Educational Robotics in Primary School and Its Possible Place in the Curricula". *Education in & with Robotics to Foster 21st-Century Skills. EDUROBOTICS 2021*. Studies in Computational Intelligence, vol 982. 2021, Springer, Cham. https://doi.org/10.1007/978-3-030-77022-8_19

[9] Bellas F. et al., "STEAM Approach to Autonomous Robotics Curriculum for High School Using the Robobo Robot", *Robotics in Education. RiE 2019. Advances in Intelligent Systems and Computing*, vol 1023. Springer, 2020. https://doi.org/10.1007/978-3-030-26945-6_8

[10] Bellas F. et al., "The Robobo Project: Bringing Educational Robotics Closer to Real-World Applications". *Robotics in Education. RiE 2017.* Advances in Intelligent Systems and Comp, vol 630. Springer, 2018, https://doi.org/10.1007/978-3-319-62875-2_20. Best paper Award.

[11] Bellas F., Salgado M., Blanco T.F., Duro R.J., "Robotics in Primary School: A Realistic Mathematics Approach", *Smart Learning with Educational Robotics*, pp 149 - 182, 2019

[12] Linda Daniela (Ed.), "Smart Learning with Educational Robots", *Springer*, 2019, https://doi.org/10.1007/978-3-030-19913-5

[13] Myint Swe Khine (Ed.), "Robotics in STEM Education", Springer, 2017, https://doi.org/10.1007/978-3-319-57786-9

[14] Ahmad Khanlari, "Teachers' perceptions of the benefits and the challenges of integrating educational robots into primary/elementary curricula", *European Journal of Engineering Education*, 41:3, 320-330, 2016, DOI: 10.1080/03043797.2015.1056106

[15] Anwar, S., Bascou, N. A., Menekse, M., & Kardgar, A., "A Systematic Review of Studies on Educational Robotics", *Journal of Pre-College Engineering Education Research (J-PEER),* 9(2), Article 2, 2019

[16] Y. A. C. González and A. G. Muñoz-Repiso, "Educational robotics for the formation of programming skills and computational thinking in childish", *2017 International Symposium on Computers in Education (SIIE)*, 2017, pp. 1-5, 2017, doi: 10.1109/SIIE.2017.8259652.

[17] Eguchi A., "Bringing Robotics in Classrooms". *Robotics in STEM Educ*. Springer, 2017. https://doi.org/10.1007/978-3-319-57786-9_1

[18] Sergey Filippov, Natalia Ten, Ilya Shirokolobov, Alexander Fradkov, "Teaching Robotics in Secondary School", *IFAC-PapersOnLine*, Volume 50, Issue 1, 2017, Pages 12155-12160, ISSN 2405-8963, https://doi.org/10.1016/j.ifacol.2017.08.2143.

[19] Jäggle G., Lammer L., Hieber H., Vincze M., "Technological Literacy Through Outreach with Educational Robotics". *Robotics in Education. RiE 2019. Advances in Intelligent Systems and Computing*, vol 1023. Springer, 2020. https://doi.org/10.1007/978-3-030-26945-6_11

[20] Amy Eguchi, "RoboCupJunior for promoting STEM education", *Robotics and Autonomous Systems*, Volume 75, Part B, 2016, Pages 692-699, ISSN 0921-8890, https://doi.org/10.1016/j.robot.2015.05.013.

[21] S. Evripidou, K. Georgiou, L. Doitsidis, A. A. Amanatiadis, Z. Zinonos and S. A. Chatzichristofis, "Educational Robotics: Platforms, Competitions and Expected Learning Outcomes", in *IEEE Access*, vol. 8, pp. 219534-219562, 2020, doi: 10.1109/ACCESS.2020.3042555.

[22] Ioannou, A., Makridou, E., "Exploring the potentials of educational robotics in the development of computational thinking: A summary of current research and practical proposal for future work", *Educ. Inf. Technol* 23, 2531-2544, 2018, DOI: 10.1007/s10639-018-9729-z

[23] Armesto, L., Fuentes-Durá, P. & Perry, D., "Low-cost Printable Robots in Education", *J Intell Robot Syst* 81, 5-24, 2016, https://doi.org/10.1007/s10846-015-0199-x

[24] A. Takacs, G. Eigner, L. Kovacs, I. J. Rudas and T. Haidegger, "Teacher's Kit: Development, Usability, and Communities of Modular Robotic Kits for Classroom Education", in *IEEE Robotics & Automation Magazine*, vol. 23, no. 2, pp. 30-39, June 2016, doi: 10.1109/MRA.2016.2548754.

[25] Tuluri F., "STEM Education by Exploring Robotics". *Robotics in STEM Education*. Springer, 2017. https://doi.org/10.1007/978-3-319-57786-9_8

[26] European Commission, "Digital Education Plan 2021-2017", 2020, Last accessed 24/06/2021: https://ec.europa.eu/education/education-in-the-eu/digital-education-action-plan_en

[27] Vuorikari, R., Kluzer, S. and Punie, Y., (2022) DigComp 2.2: The Digital Competence Framework for Citizens, EUR 31006 EN, Publications Office of the European Union, Luxembourg, 2022, ISBN 978-92-76-48882-8, doi:10.2760/115376, JRC128415.

[28] Ng, D., Leung, J., Chu, S., Qiao, M. (2021) Conceptualizing AI literacy: An exploratory review, Computers and Education: Artificial Intelligence, Volume 2, https://doi.org/10.1016/j.caeai.2021.100041.

[29] Jeremy Knox. "Artificial intelligence and education in China", *Learning, Media and Technology*, 45:3, 298-311, 2018, DOI: 10.1080/17439884.2020.1754236

[30] UNESCO (2022). K-12 AI curricula: a mapping of government-endorsed AI curricula. https://unesdoc.unesco.org/ark:/48223/pf0000380602.locale=en. Accessed March 2022.

[31] Russell, S. & Norvig, P., "Artificial Intelligence. A Modern Approach". *Pearson*, 2021

[32] AI4K12 initiative, Last accessed 24/06/2021: https://ai4k12.org

[33] F. Bellas, S. Guerreiro-Santalla, M. Naya, M., R.J. Duro, "AI Curriculum for European High Schools: An Embedded Intelligence Approach". International Journal of Artificial Intelligence in Education, 2022. https://doi.org/10.1007/s40593-022-00315-0

[34] Siu-Cheung Kong, Ming Ming Chiu, Ming Lai, "A study of primary school students' interest, collaboration attitude, and programming empowerment in computational thinking education", *Computers & Education*, Volume 127, pp 178-189, 2018

[35] Pedersen B.K.M.K., Larsen J.C., Nielsen J., "The Effect of Commercially Available Educational Robotics: A Systematic Review". *Robotics in Education. RiE 2019. Advances in Intelligent Systems and Computing, vol 1023*. Springer, 2020. https://doi.org/10.1007/978-3-030-26945-6_2

[36] Ribeiro, A., Lopes, G. "Learning Robotics: a Review". *Curr Robot Rep* 1, 1-11, 2020. https://doi.org/10.1007/s43154-020-00002-9

[37] P. Voštinár, "Using mBot robots for the motivation of studying computer science," *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2020, pp. 653-657, doi: 10.23919/MIPRO48935.2020.9245308.

[38] Laurens Valk. 2014. The LEGO MINDSTORMS EV3 Discovery Book: A Beginner's Guide to Building and Programming Robots (1st. ed.). No Starch Press, USA. ISBN: 1593275323

[39] F. Riedo, M. Chevalier, S. Magnenat and F. Mondada, "Thymio II, a robot that grows wiser with children," *2013 IEEE Workshop on Advanced Robotics and its Social Impacts*, 2013, pp. 187-193, doi: 10.1109/ARSO.2013.6705527.

[40] Digital Dream Labs, "Cozmo", digitaldreamlabs.com, https://www.digitaldreamlabs.com/pages/cozmo (accessed September 26, 2022)

[41] Learningrobots, "Alphai", learningrobots.ai, https://learningrobots.ai/?lang=en (accessed September 26, 2022)

[42] M. Pacheco, R. Fogh, H. H. Lund and D. J. Christensen, "Fable II: Design of a modular robot for creative learning," 2015 *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 6134-6139, doi: 10.1109/ICRA.2015.7140060.

[43] Amsters, R., & Slaets, P. (2019, April). Turtlebot 3 as a robotics education platform. *International Conference on Robotics in Education (RiE)* (pp. 170-181). Springer, Cham. https://doi.org/10.1007/978-3-030-26945-6_16

[44] Soares, J. M., Navarro, I., & Martinoli, A. (2016). The Khepera IV mobile robot: performance evaluation, sensory data and software toolbox. *Robot 2015: second Iberian robotics conference* (pp. 767-781). Springer, Cham. https://doi.org/10.1007/978-3-319-27146-0_59

[45] Shamsuddin, S., Ismail, L. I., Yussof, H., Zahari, N. I., Bahari, S., Hashim, H., & Jaffar, A. (2011, November). Humanoid robot NAO: Review of

control and motion exploration. *2011 IEEE international conference on Control System, Computing and Engineering* (pp. 511-516). IEEE. doi: 10.1109/ICCSCE.2011.6190579.

[46] Farley, A., Wang, J., & Marshall, J. A. (2022). How to pick a mobile robot simulator: A quantitative comparison of CoppeliaSim, Gazebo, MORSE and Webots with a focus on accuracy of motion. *Simulation Modelling Practice and Theory*, 120, 102629. https://doi.org/10.1016/j.simpat.2022.102629

[47] Gazebo, "Gazebo", gazebosim.org, https://classic.gazebosim.org/tutorials?tut=guided_b1&cat= (accessed September 26, 2022)

[48] Ketterl, M., Jost, B., Leimbach, T., & Budde, R. (2016). Tema 2: Open roberta-a web based approach to visually program real educational robots. *Tidsskriftet Læring og Medier (LOM)*, 8(14). doi: https://doi.org/10.7146/lom.v8i14.22183

[49] Robotbenchmark.net, https://robotbenchmark.net/ (accessed September 26, 2022)

[50] Llamas L.F., Paz-Lopez A., Prieto A., Orjales F., Bellas F., "Artificial Intelligence Teaching Through Embedded Systems: A Smartphone-Based Robot Approach". *Robot 2019: Fourth Iberian Robotics Conference. ROBOT 2019. Advances in Intelligent Systems and Computing*, vol 1092. Springer, 2020. https://doi.org/10.1007/978-3-030-35990-4_42

[51] J. Tang, et al., "Teaching autonomous driving using a modular and integrated approach", in 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), 2018, vol. 1, pp. 361-366.

[52] Autoauto platform. Last accessed 24/06/2021. https://www.autoauto.ai

[53] V. Costa, R. Rossetti, and A. Sousa, "Simulator for teaching robotics, ROS and autonomous driving in a competitive mindset," Int. J. Technol. Hum. Interact., vol. 13, no. 4, pp. 19-32, 2017.

[54] Tani J. et al., "Duckietown: An Innovative Way to Teach Autonomy". *Educational Robotics in the Makers Era. Edurobotics 2016*. Advances in Intelligent Systems and Computing, vol 560. Springer, 2017. https://doi.org/10.1007/978-3-319-55553-9_8

[55] Tselegkaridis, Sokratis, and Theodosios Sapounidis, "Simulators in Educational Robotics: A Review" *Education Sciences*11, no. 1: 11, 2021, https://doi.org/10.3390/educsci11010011

[56] Christina Chalmers, "Robotics and computational thinking in primary school", *International Journal of Child-Computer Interaction*, Volume 17, pp. 93-100, 2018

[57] S. Papavlasopoulou, M. N. Giannakos and L. Jaccheri, "Reviewing the affordances of tangible programming languages: Implications for design and practice," 2017 IEEE Global Engineering Education Conference (EDUCON), 2017, pp. 1811-1816, doi: 10.1109/EDUCON.2017.7943096.

[58] Sapounidis T., Demetriadis S., "Educational Robots Driven by Tangible Programming Languages: A Review on the Field". *Educational Robotics in the Makers Era. Edurobotics 2016*. Advances in Intelligent Systems and Computing, vol 560. Springer, 2016. https://doi.org/10.1007/978-3-319-55553-9_16

[59] M. Naya et al., "A versatile robotic platform for educational interaction," 2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2017, pp. 138-144, doi: 10.1109/IDAACS.2017.8095065.

[60] Mint for People, "Robobo project", theroboboproject.com, https://theroboboproject.com/en/ (accessed September 26, 2022)

[61] Maloney, J., Resnick, M., Rusk, N., Silverman, B., & Eastmond, E. (2010). The scratch programming language and environment. *ACM Transactions on Computing Education (TOCE)*, 10(4), 1-15. doi https://doi.org/10.1145/1868358.1868363

[62] Mint for People, "RoboboSim", github.com/mintforpeople/robobo-programming, https://github.com/mintforpeople/robobo-programming/wiki/Unity (accessed September 26, 2022)

[63] Quigley, M.; Conley, K.; Gerkey, B. P.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R. & Ng, A. Y, (2009, May). ROS: an open-source Robot Operating System. *ICRA workshop on open source software* (Vol. 3, No. 3.2, p. 5).

[64] Mint for People, "Robobo programming wiki", github.com/mintforpeople, https://github.com/mintforpeople/robobo-programming/wiki (accessed September 26, 2022)

[65] E. A. Esquivel-Barboza, L. F. Llamas, P. Vázquez, F. Bellas and E. Arias-Méndez, "Adapting Computer Vision Algorithms to Smartphone-based Robot for Education", *2020 Fourth International Conference on Multimedia Computing, Networking and Applications (MCNA)*, 2020, pp. 51-56, doi: 10.1109/MCNA50957.2020.9264289.

[66] Xunta Galicia, "STEMbach course 2019/2020", edu.xunta.gal, https://www.edu.xunta.gal/portal/es/node/27960 (accessed September 26, 2022)

[67] Juanatey, D.; Naya, M.; Baamonde, T.; Bellas, F. Developing a Simulation Model for Autonomous Driving Education in the Robobo SmartCity Framework. *Eng. Proc*. 2021, *7*, 49. https://doi.org/10.3390/engproc2021007049

Martin Naya Varela received a B.S. in industrial engineering from the University of Coruña in 2013 and a Ph.D. in 2021 from the same university. He is currently working as a postdoc at the Integrated Group for Engineering Research, at the University of Coruña, Spain. His research interests include the development of educational robotics and AI fields, such as developmental robotics, embodied cognition and evolutionary algorithms.

Sara Guerreiro-Santalla received a B.S. in electronic engineering from the University of Coruña in 2018 and a McC. in education in 2019. She is currently a predoc researcher at the Integrated Group for Engineering Research, at the University of A Coruña, Spain. Her research interests are focused on teaching innovation, mainly in Artificial Intelligence for Education.

Tamara Baamonde has a B.S. in Industrial Design from the University of Coruña. She is currently working as a technician and industrial designer at the Integrated Group for Engineering Research, at the University of Coruña, Spain. She is also member of a European project called AI+, whose aim is the creation of an Artificial Intelligence curriculum for pre-universtiy students units. Her main interests are education, robotics and interface design.

Francisco Bellas is a Full Professor at the University of Coruña, Spain. He received the B.S. and M.S. degree in Physics from the University of Santiago de Compostela, Spain, in 1999, and a PhD in Computer Science from the University of Coruña in 2003. He is a member of the Integrated Group for Engineering Research at the University of Coruña. His main research activities are focused to cognitive robotics, collective intelligence, educational robotics and artificial intelligence education.