An Open Source Framework for Educational Applications Using Cozmo Mobile Robot

Victor Luis Pires Kusumota
Electrical Engineering Department
Federal University of São Carlos
São Carlos, Brazil
victor.kusumota@hotmail.com

Rafael Vidal Aroca
Mechanical Engineering Department
Federal University of São Carlos
São Carlos, Brazil
rafaelaroca@ieee.org

Felipe Nascimento Martins
Institute of Engineering
Hanze University of Applied Sciences
Assen, Netherlands
fe.nascimento.martins@pl.hanze.nl

***Abstract*—Cozmo is a real-life robot designed to interact with people playing games, making sounds, expressing emotions on a LCD screen and many other pre-programmable functions. We present the development and implementation of an educational platform for Cozmo mobile robot, with several features, including web server for user interface, computer vision, voice recognition, robot trajectory tracking control, among others. Functions for educational purposes were implemented, including mathematical operations, spelling, directions, and questions functions that gives more flexibility for the teachers to create their own scripts. In this system, a cloud voice recognition tool was implemented to improve the interactive system between Cozmo and the users. Also, a cloud computing vision system was used to perform object recognition using Cozmo's camera, to be applied on educational games. Other functions were created with the purpose of controlling the emotions and the motors of Cozmo to create more sophisticated scripts. To apply the functions on Cozmo robot, an interpreter algorithm was developed to translate the functions into Cozmo's programming language. To validate this work, the proposed framework was presented to several elementary school teachers (classes with students between 4 and 12). Students and teacher's impressions are reported in this text, and indicate that the proposed system can be a useful educational tool.**

***Index Terms*—Web server, educational robot, computer vision, educational games**

I. Introduction

The use of robotics is becoming increasingly present at all levels of education, including elementary, middle, undergraduate and postgraduate education. Nowadays, it is known that robotics, as a multidisciplinary tool, is an ideal solution for curricular integration, offering students the possibility to relate concepts learned in different disciplines [1]–[3], also promoting, naturally, the incentive to work in teams [2], [4]. In addition, most people learn more easily when practical activities are involved in the learning process [5]. Therefore, robots can be used as pedagogical tools that offer a "learn by doing" experience [6], [7].

According to the American philosopher John Dewey, a human being learns through a 'hands-on' approach [8]. Dewey believes that real-life experiences are very important for students to learn and adapt to their environment. According to the study [8], "Dewey's views have had a profound impact on educational systems. They have provided the philosophical basis for learning by doing, project work, simulation and many forms of experiential education" [8].

However, there are still some problems for the wider adoption of robotics technologies in education, the main ones being costs and difficulty of use. Regarding costs, several authors [3], [6] warn that the high costs of robots can prevent their use in classrooms. Other authors have proposed software tools to simulate robots to provide students access to educational robotics without the need of real robots [9], [10].

The development of this project is motivated by the fact that, to the best of our knowledge, there is no correlative literature in educational robotics that gives freedom and flexibility to teachers to develop their own proposals and didactic scripts for robots to interact with students. The existing works consist of actions previously defined and programmed in a fixed way. In this new development, the scripts made by teachers in the Web system are automatically made available to robots connected to this platform, allowing different teachers to implement different activities.

II. Related Work

Jeonghye Han from Cheongju National University of Education has established that there are two types of educational robots: hands-on robots and educational service robots. On his research, Han consider that hands-on robots are used for sciences, technology, engineering and mathematics education to increase people's creativity and interest for those fields [11]. For the educational service robots, Han argues that because of their friendly appearance and physical movements, student's interests increases when the robots interact with them, making learning more fun [11].

In [12], Gu-Min Jeong and his colleagues argue that when the interaction between the robot and child is natural, the children's learning presents betters results. Thus, the robot system has to be sophisticated and intelligent to be able to interact with children naturally. Because of this, the system costs increases, which currently is a problem for using a robot-based learning system [10], [12].

A research performed at the Federal University of Rio Grande do Norte (UFRN), Brazil, presented ways of developing very low cost robots that can be used in educational

Project developed with financial support from the São Paulo Research Foundation (FAPESP).

environments. A so-called "N-Bot" robot was developed and uses an audio channel as a control interface. The frequencies of these audios are used both to transmit commands to the actuators and to receive the signals captured from the N-Bot's sensors [13]. These signals can be interpreted by a control computer, a smartphone or a tablet, which must be accessed and programmed remotely through local networks or the Internet, or cloud robotics services can be used. In addition, it is also possible to use a Web page, where students can create programs for the audio's robot generation and reception in order to apply controls [13]. N-bot's image recognition system was developed using the mobile device's camera installed in the robot and using a web programming, one can see the images captured by the camera, on real time [13], through a web browser.

A work performed by Cibele Alves and her colleagues presents that the difficulty of students in learning logic and programming languages provides several technological solutions to help the students learn how to program [14]. On this work, two approaches were used: robotics and programming from graphical languages [14]. The literature shows that using this tools, it is possible to help young people to develop a so called "computational thinking" and make them think in a systematic way [14]. Thus, a research performed at the Universidade Tecnológica Federal do Paraná (UTFPR), Brazil, proposes a platform development that consists of a Web programming environment in blocks, and a smartphone as a robotic base controlling unit. The Model View Controller (MVC) programming model was used. In the MVC, the Controller receives user requests, Model accesses the database using the data inserted by the user in the request and a View, for example a HTML page, is returned to the user [14].

III. Methods

A. Cozmo Robot

Cozmo robot is a toy robot, developed by the Anki Company, designed to interact with people playing games, making sounds and expressing emotions on a LCD screen. Cozmo is an intelligent tracked robot that includes four motors, fifty gears, a 30 fps VGA camera with facial recognition software, a 128x64 resolution facial display, a speaker to communicate and an arm to play games.

Anki provides an application called "Cozmo", which can be downloaded on Google Play Store. This application is used to give functionality to the robot, since it is necessary to run it to start the robot. In this application, there is a set of tools that make it possible to program Cozmo called Cozmo SDK.

B. Cozmo SDK

The Cozmo SDK is an open source platform based on Python language. With a comprehensive set of low- and high-level functions, and full access to sensor data, the SDK is as simple or as granular as a person needs it to be [15].

Cozmo communicates with the smartphone through a Wi-Fi network created by the robot itself. Therefore, it is necessary to start the application "Cozmo" created by the company Anki and connect the smartphone to the Cozmo's network. Then, the application provides a function called "enable SDK". When selecting such function, it is possible to transfer algorithms written in Python language to be executed by Cozmo. To transfer the algorithms, it is necessary to connect the smartphone to the computer via USB port and then, run the ".py" algorithm file. Fig. 1 shows the communication schematic module between Cozmo, the smartphone and the computer.

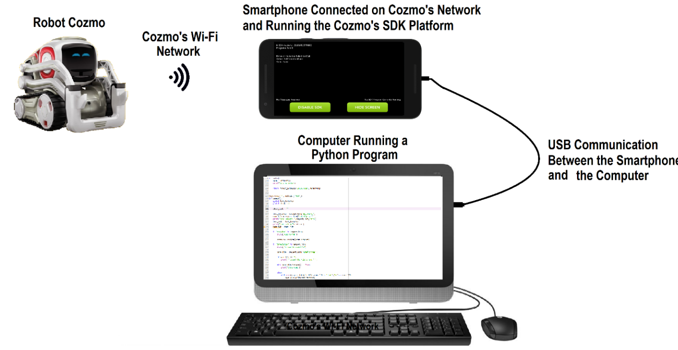

Fig. 1. Communication between the robot Cozmo, the smartphone and the Computer.

The SDK module installation process depends on the devices used. For this work, it was used a computer running Windows 10 and a smartphone with a 8.0.0 Android platform version. The files to download and all the information needed for the Windows installation can be found in Anki's website [16].

C. Voice Communication System

This feature was implemented to make the interaction system with Cozmo more sophisticated and realistic.

As Cozmo does not have a built-in microphone, it was necessary to use the computer's microphone to capture the environment sound voices. In this project, it was decided to use the Google Cloud Speech API [17]. This tool provides ease implementation, availability of over 110 languages to perform the text recognition and the ability to return the results in real time. In Google Cloud Speech operation, the sounds captured by the computer's microphone are sent to a cloud server, where the speech recognition is performed and then, returns the text results to be used in the algorithm.

D. Computer Vision System

The service used in the method is the Google Cloud Vision API [18]. This service receives images through web requests and returns a set of image textual characteristics. The system can recognize any type of image and identify predominant objects such as cars, people, faces and people mood (sad, happy, angry, etc) as well as detecting texts, places and logos.

As Cozmo has a 30 fps VGA built-in camera, an algorithm was developed to take pictures and perform the image recognition using the Google Cloud API. This API service was used to perform 2 types of vision recognition: Text and object.

The object recognition was implemented for Cozmo to identify specific objects during a orientation game, which is discussed on the Software Development section. Using the text recognition feature, it is possible to extract all the texts in a image and the individual words, with their bounding boxes pixel positions. The implementation of this feature is discussed on the Software Development section.

E. Trajectory Controller System

The controller used on this work was presented in [19]. It uses a (x, y) desired coordinate positions as the setpoint to control the robot angular and linear speeds to follow the given trajectory. Using Cozmo's encoders to measure it's real coordinates and angular position (the process variables), it is possible to implement an algorithm to calculate the linear and angular speeds to be applied to the motors. It is worth mentioning that this controller uses the center position of Cozmo's arm as the point which position is controlled.

For implementing the controller, the linear and angular speeds are calculated using (1).

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} cos(\varphi) & sin(\varphi) \\ \dfrac{-1}{\alpha} sin(\varphi) & \dfrac{1}{\alpha} cos(\varphi) \end{bmatrix} \begin{bmatrix} dxd + k_x x_{err} \\ dyd + k_y y_{err} \end{bmatrix}, \tag{1}$$

where:

- v and ω are the linear and angular speeds, respectively;
- φ is the robot orientation angle with respect of X axis;
- α is the distance between the robot axle's center and the robot arm's center;
- dxd and dyd are the robot x-axis and y-axis speeds, respectively;
- k_x and k_y are the x-axis and y-axis controller gains, respectively;
- x_{err} and y_{err} are the difference between the actual and desired x and y coordinates of the robot, respectively.

The values used on the controller are: α = 40 mm, k_x = 0.6 and k_y = 0.6. It is worth to mention that as Cozmo's encoder sensors return the measured values in millimeter, all the controller variables are also calculated in the millimeter, as well as the constant parameters. Equations (2) and (3) show the functions used to calculate the desired speeds of right and left motors speeds, respectively.

$$v_{right} = v + \frac{d}{2}\omega, \tag{2}$$

$$v_{left} = v - \frac{d}{2}\omega, \tag{3}$$

where d is the distance between the left and right wheels. For Cozmo robot, the distance measured is d = 44.8 mm.

After applying the left and right motor's speed, it is necessary to measure the new actual values of x, y positions, and the φ orientation of Cozmo. Measuring those values, the controller algorithm starts again. This process is repeated in a loop until the robot reaches the end of the path.

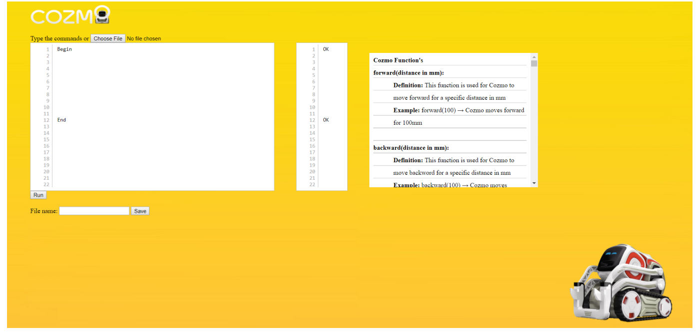

Fig. 2. Web page where the user writes a sequence of commands to be sent to the robot. The left white area is the space for the commands, the middle text area is the code checker where it says if the codes written is right or wrong and the right area shows an explanation of all commands.

F. Web Page

To implement the Web server system on the Cozmo SDK platform, it was necessary to develop a web page using the Hypertext Markup Language HTML.

In an HTML page it is possible to implement several types of elements and functions. To write scripts to be sent to the Web server and to indicate the user if there are errors in the script, two textarea elements were used. To give instruction of how to implement the functions, another textarea element was used.

A function was implemented to allow users to save the scripts elaborated in the computer internal memory connected to the Web server and also load the scripts that are already stored in the computer's storage. In order to make the web page visually interesting, an image of background and figures referring to the robot Cozmo were used. Those images were downloaded from Anki's website and modified by the project developer. Fig. 2 shows the developed Web page.

G. Cozmo's Pen Holder Design

For Cozmo to draw figures on a surface, it was necessary to design a drawing pen holder part. Solid Works modeling software was used to design an arm piece with a hole that allows to attach a drawing pen on Cozmo's front lift. Fig. 3 shows the designed piece images from different angles.

By attaching the 3D printed piece to Cozmo's arm, it is possible to develop movement algorithms to draw geometric forms. Fig. 4 illustrates Cozmo with the drawing piece and the pen attached.

H. Software Development

The Software development consists of implementing the Web server using the Cozmo SDK platform.

In this project, it was used a web framework called Flask. Once imported into the algorithm, Flask can be used to save time by building web applications [20]. Because of the Jinja2 Flask library, it is possible to use the `render_template` function to render a previously created HTML page.

In the Web Server algorithm developed, functions that perform the transfer of data between a client (user) and

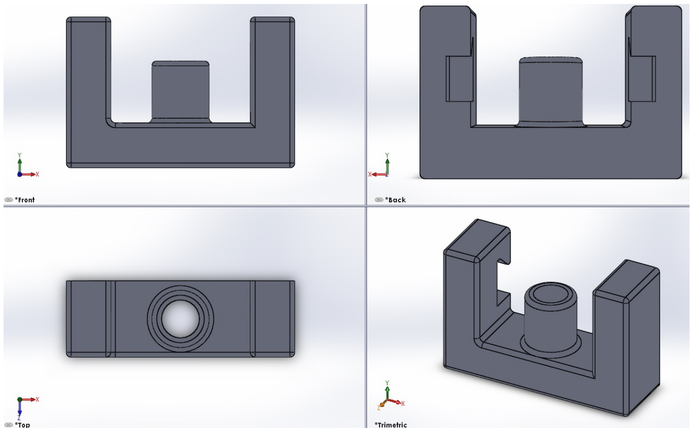

Fig. 3. Arm piece structure design to attach a drawing pen on Cozmo.

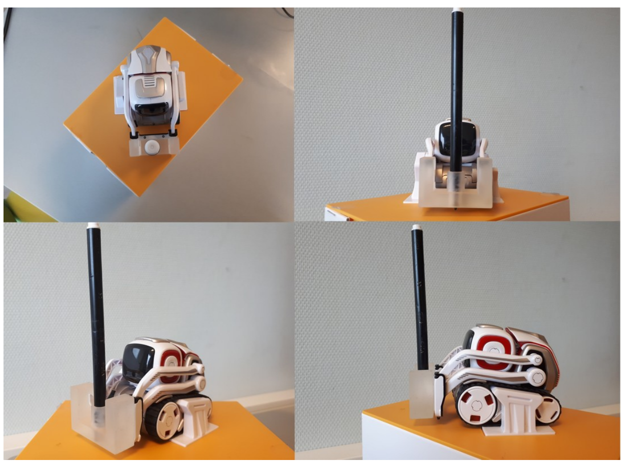

Fig. 4. Arm piece structure design to attach a drawing pen on Cozmo.

the server were implemented. A Hypertext Transfer Protocol (HTTP) request method was implemented.

An algorithm was used to extract text information from textarea element from the rendered ".html" file and print such information to the user. Then, it is possible to process the data sent by the client and to elaborate an text algorithm interpreter. The interpreter's function is translating the script line by line into the Python language and run commands on Cozmo.

1) Mathematical Functions: Four mathematical functions have been developed for helping students to learn: sum, subtraction, division and multiplication. In the sum command, the programmer must specify the two numbers to be added. This command must be implemented as follows: sum ($n1$ + $n2$), where $n1$ and $n2$ represent the numbers to be summed. An algorithm was programmed to activate the robot's speakers to say "How much is $\{n1\}$ + $\{n2\}$?". The user interacting with Cozmo should find the solution and give the answer via voice communication, which uses the Google Cloud Speech API tool, as discussed earlier. Then, the answer given by the user is checked. Cozmo reacts with a happy or sad behavior, depending on the answer. If it is wrong, Cozmo gives the correct answer.

The subtraction, division and multiplication commands work similarly to the sum command. The difference is the commands implementation, which are shown below:

- Subtraction: sub($n1$ - $n2$)
- Multiplication: mult($n1$*$n2$)
- Division: div($n1$/$n2$)

2) Drawing Algorithm: During the early school years, students are taught some basic concepts about geometric shapes. The purpose of teaching geometry is related to the sense of localization, figure recognition, manipulation of geometric forms, spatial representation and establishment of properties [21]. Because of this, an algorithm was developed to make Cozmo draw 4 types of geometric shapes and ask questions about the shapes characteristics.

An algorithm was developed to make Cozmo draw 4 geometric shapes: Square, triangle, circle and rectangle. The command to implement the drawing functions on the Web Page is "draw", indicating the geometric shape (square,triangle, circle or rectangle). For example: draw square.

Using the same methods as the drawing function, it was created a game of questions about the geometric figures drawn by Cozmo. The command to implement this functions is "game", indicating the desired geometric shape. For each shape, the following questions are asked by Cozmo:

- Square:
  - What is the name of this geometric shape?
  - How many lines does a square shape have?
  - How many equal lines does a square shape have?
- Circle:
  - What is the name of this geometric shape?
  - Is a circle's shape round?
  - Does a circle have straight lines?
- Triangle:
  - What is the name of this geometric shape?
  - How many lines does a triangle shape have?
  - Is a triangle's shape round?
- Rectangle:
  - What is the name of this geometric shape?
  - How many lines does a rectangle shape have?
  - Is a rectangle shape the same as a square shape?

So when Cozmo finishes drawing, all the 3 questions are asked to the person that the robot is playing with, and the answer must be given by voice command. In case the person gets the correct answer for the question, Cozmo plays a happy animation and congratulates the person. Otherwise, Cozmo plays a sad animation, says the correct answer, and tells to look at the computer screen to watch a video to understand more about the drawn geometric shape.

To give more flexibility for the programmers to draw other types of figures, an algorithm module was created for drawing on a computer screen using the mouse cursor and then, the trajectory controller is used for Cozmo to draw the same figure on a surface.

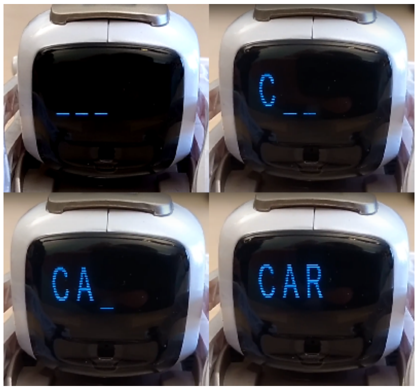

Fig. 5. Sequence of Cozmo's LCD screen for spelling the word "car".

3) Spell Function: It is known that during the early years of elementary school, students are taught how to identify and pronounce their native language alphabet letters. This knowledge is the basis of knowledge for students to learn how to read and write texts.

Because of this, a command called "spell word ()" was created to encourage learning the Latin alphabet. The teacher must indicate a word to be spelled by the student. The characters "_" are displayed on the LCD screen of Cozmo indicating the letters that have yet to be spelled. As the student spells the letters, the characters "_" are replaced by the spelled letters. Fig. 5 shows the image of Cozmo's LCD screen for an example of implementing the function spell word, with the desired word being "car" (spell word (car)).

4) Read Text: To encourage and help students to learn reading, a function "read text" was created. This function uses Cozmo's camera to take a picture and implement the computer vision algorithm to make a text recognition. Then, Cozmo activates the speakers to say the recognized words. To use this function, it is necessary to put the desired text to be read in front of Cozmo and run the command "read text".

5) Direction Function: In order to help the process of teaching direction, another game was created. This game consists in giving Cozmo directions to find a desired object specified by the programmer. The student needs to speak the directions forward, backward, left or right to control Cozmo's movement towards the specified object. When Cozmo is in front of the object, it is necessary to speak "picture" to activate Cozmo's camera, take a picture, and use the computer vision system to make a object recognition. If the found object is correct, Cozmo's plays the happy animation. Otherwise, Cozmo plays the sad animation.

6) Control Functions Module: A function module was created that guarantees a greater programming flexibility to teachers, allowing the control of several types of actions. This module consists of functions that allow control of the wheel motors, arm motor, speaker, Cozmo animations, and also display videos on the computer screen. The commands created are shown below.

- forward(Distance in millimeters)
- backward(Distance in millimeters)
- left(Angle in degrees)
- right(Angle in Degrees)
- pen up
- pen down
- head up
- head down
- play animation (Type of emotion)
- play video (Youtube video link)
- speak(Words to speak)
- ask (Question to be asked)
- wait for answer
- if answer = Correct answer
- end if
- if not
- end if not

The functions that control the movement of the robot were based on a Python educational library named Turtle Graphics. This library was designed to control a virtual pen movement on the computer screen, so it is possible to draw all types of figures.

The command play animation (Type of emotion) was created with the purpose of making Cozmo's interaction system more fun. It makes use of Cozmo's built-in animations to make Cozmo reacts with a specific mood. The play video command was created to play a YouTube video on the computer screen, as the same way it does on the geometric figure game mentioned before. In order to control Cozmo's speech, the "speak" function was created for Cozmo to speak the words typed between the command parentheses. However, not every language is currently available. Only the languages English, French, German and Japanese are supported.

The main command that enables the creation of scripts and allows the application of the other functions discussed in this section is the "ask" command. This command must be implemented together with the "wait for answer", "if answer = (Correct answer)" and "if not". The "ask" command allows the teacher to make Cozmo ask any type of question (including mathematical problems questions). The command "wait for answer" must be used to activate the recognition speech system to identify the student's response.

The "if answer = Correct answer" command works similar as other computer languages(for example C, C++ and Python). The teacher must type the question's correct answer. So if the answer given by the student is the same as the answer typed by the teacher, the algorithm runs the "if answer = Correct answer" conditional structure. Otherwise, the algorithm runs the "if not" conditional structure. Using those commands, it is possible to program Cozmo with all the functions mentioned in this section, so the teachers can create different types of scripts and control the speech, animations, the robot movements and also display videos on the computer screen, depending on the answer given by the student. The commands "end if" and "end if not" are used to indicate when the conditional structures of "if answer = Correct answer" and "if not" ends, respectively. Fig. 6 shows an example of the ask function.

```
Begin


ask(How many letters does the Latin alphabet have?)

wait for answer

if answer = 26
play animation (happy)
speak(Congratulations! You are right!)

end if

if not
play animation (sad)
speak(Sorry, that is not the right answer. The right answer is 26)
speak(Look at the computer screen to watch a video about the alphabet)
play video (https://www.youtube.com/watch?v=Sw2KZki-eaA)


End


```

Fig. 6. Script with the ask function example with the the correct implementation.

The source code and application is available at: https://github.com/viclpk/CozmoEducationalTool

IV. Results and Discussion

As previously discussed, 4 geometric shapes were chosen to program Cozmo to draw: Circle, square, triangle and rectangle. The YouTube links below show videos of Cozmo drawing each figure:

- Square: https://youtu.be/NJYDW_IDxvI
- Circle: https://youtu.be/h8RV66g1-KQ
- Triangle: https://youtu.be/RvKaNv7ySgg
- Rectangle: https://youtu.be/KM4UMc-H6Pk

Fig. 7 illustrates the geometric shapes drawn using the Cozmo's draw functions, corresponding to the above videos.

Because Cozmo has a sophisticated motor controller, the drawn geometric figures presents good results, as it can be seen in the Fig. 7. It is worth to mention that the drawing accuracy is important for the students to identify and memorize the shapes correctly.

To illustrate an example of the geometric figures asking question game, the circle game function was implemented and filmed. The video can be watched on the following YouTube link video: https://youtu.be/LLKLJMn_zwA. On this video, it is possible to see how Cozmo reacts with the correct answers given by the student and it can be observed that the interaction system becomes more natural and enthusiastic when Cozmo's emotions are used.

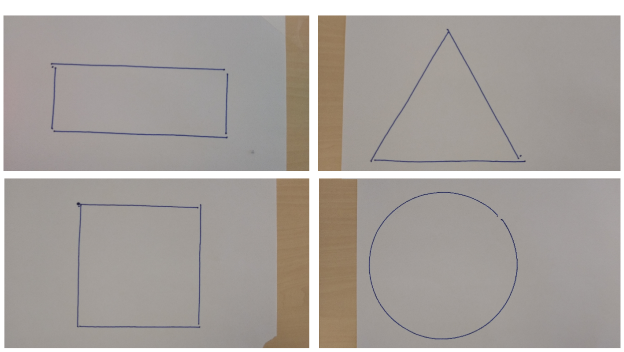

Fig. 7. The 4 geometric shapes drawn by Cozmo.

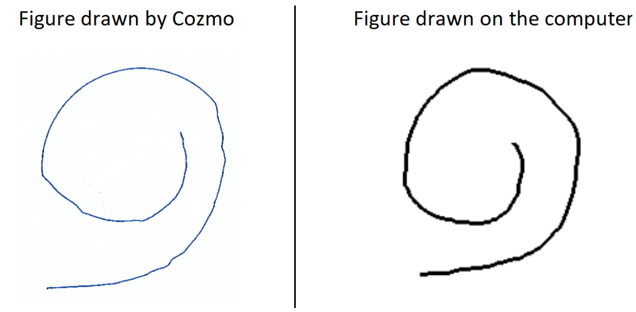

Fig. 8. Comparison between a figure drawn on the computer and the figure drawn by Cozmo using the trajectory tracking controller developed.

Also, a trajectory controller example video was created to show Cozmo drawing the desired figure drawn on the computer screen using the mouse cursor. The video is available on the following YouTube link: https://youtu.be/Nbn1C86MA4c. Fig. 8 shows another drawing example.

To demonstrate the mathematical games functionality, 2 tests were performed: sum and subtraction problems games. In the sum game, it was given the correct answer, and in the subtraction game, the wrong answer. Those tests were performed to demonstrate Cozmo's reactions for both situations. The YouTube link videos are shown below:

- Sum: https://youtu.be/7zt15YOWyOQ
- Subtraction: https://youtu.be/XOnCyyLitnI

A spelling word example video was also filmed. On this video, the desired word to be spelled is "car". The YouTube link video is available in: https://youtu.be/EGiHt7ipJvY.

Tests were performed with the computer vision system. The object recognition system was tested with 2 objects: a baseball ball and a passion fruit. Fig. 9 and Fig. 10 and show the results.

As it can be seen on Fig. 9 and Fig. 10, the ball was successfully recognized, but the passion fruit was identified as a sphere. The results indicates that the computer vision system developed could fail on some objects.

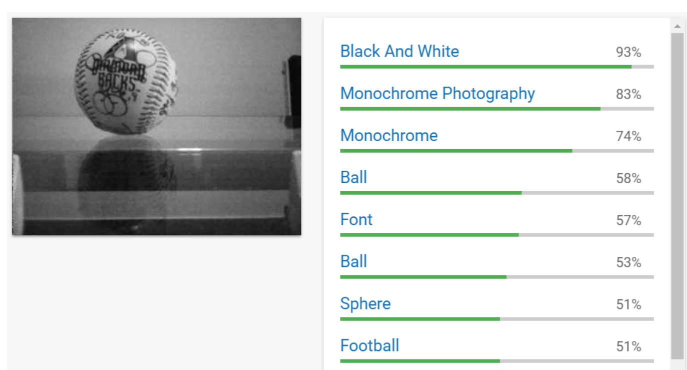

Fig. 9. Object recognition test to recognize a baseball ball using Google Cloud Vision.

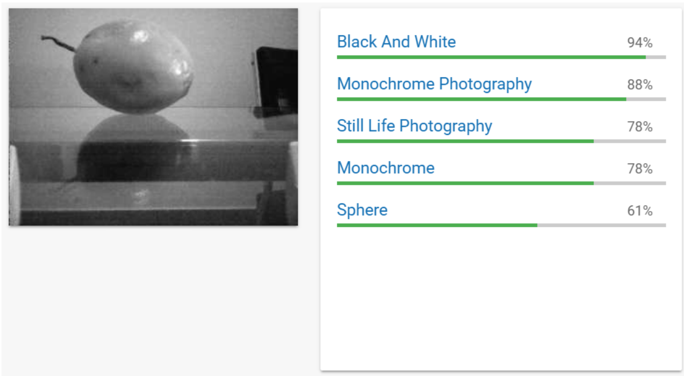

Fig. 10. Object recognition test to recognize a passion fruit using Google Cloud Vision.

The last computer vision tested was the text recognition system. The test was performed on an image with big and small letters. Fig. 11 and Fig. 12 presents a photo taken by a Smarphone camera and the test results using Cozmo's camera.

Because of Cozmo's low resolution camera, only the bigger letter words could be recognized on the test.

During the performance tests, some problems were verified on the connection with the Web server. Testing the same functions several times, sometimes the developed system presented errors, not running the function on Cozmo.

To validate the developed system, several tests were performed with two voluntary elementary school teachers, one elementary school class and one educational robotics teacher. In the tests, a briefly explanation was given for the teacher to understand the purpose of the project and how to apply the project in the classrooms. Also, a questionnaire with several questions about the project was given to the teachers. The first school to participate on the tests was the CBS Compass in Assen.

At CBS Compass school, tests were performed only with the teacher. After the tests, questions were asked to evaluate the system performance. The teacher was not familiar with any programming language but when asked about the difficulty

Fig. 11. Image tested photo taken by a Smartphone camera.

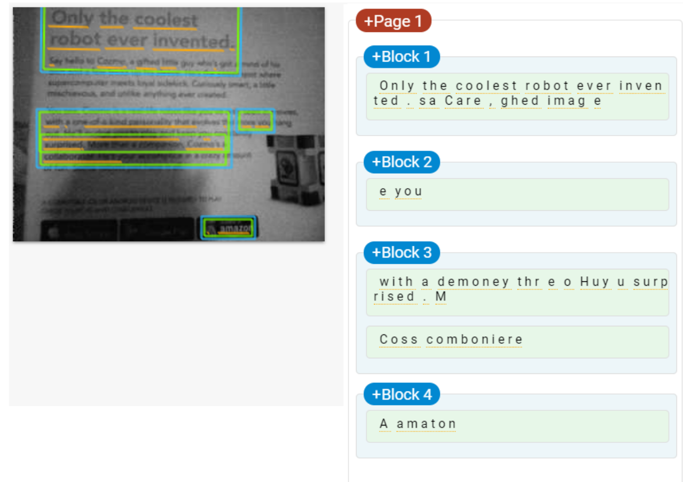

Fig. 12. Text recognition test using Cozmo's camera.

of the created language, she answered that it is not hard to understand. The teacher pointed out that she would use the developed project as a teaching tool for her classes. However, she thinks that it could be a problem that Cozmo speaks just a few languages, so the project could be applicable only for schools in which students speak the same languages as Cozmo.

The second test was performed with the educational robotics teacher, who uses Ozobot to teach basic programming concepts for students between 8 and 12 years old. As the teacher was familiar with programming, the project explanation was more technical and short. Also, during the tests, the teacher pointed out that all the students get really enthusiastic working with the Ozobots robots and that makes all the learning process easier.

The last test was performed on Emmaschool, in Assen, with a teacher and a classroom with students between the ages 10 and 12. Firstly, a Cozmo presentation was made for the class, so that students could get familiar with the robot. The first function tested was the drawing shapes function. Cozmo was programmed to draw a circle and a square shapes. During the drawing process, it was noted that all the students were focused only on Cozmo. After Cozmo finished the drawing, the students showed a surprised reaction and applauded the robot. The second function tested was the sum function. Volunteers were requested and most of students showed interested in participating and playing with Cozmo. A student was chosen by the teacher and the Cozmo was programmed to ask for the solution of "5+1". The student got the right answer and Cozmo played the happy animation. The student showed a pleased reaction and was really joyful to be playing with the robot. The last function tested was the spelling word. Another volunteer student was chosen to test the function. The word to be spelled was the student's name, which was "Anne". The student spelled the "A","n" and n" letters right, but because of she was speaking low, the voice recognition system identified the spoken letter "e" as "a", so Cozmo played the sad animation to show that it was the wrong letter.

Because of the short time available to perform the tests with the class, it was not possible to test all the developed functions with the students.

After testing with the class, the functions were showed to the teacher and explained how to applied those functions on Cozmo. Then, the teacher answered the questionnaire, and pointed out that she believes that all mathematical functions are useful for her classes. Also, the teacher could easily understand the programming language developed.

The questionnaire's answers of the three teachers can be found on the following Google Drive link: drive.google.com/file/d/14ZPXOfrHasZjf1cj6vDq0lhTtTJpjAXJ/view?usp=sharing

V. Conclusion

In this work, a Web server System with cloud architecture, voice recognition and computer vision system was developed to run educational functions and games in a robot called Cozmo.

The tests performed with the teachers presented positive results. The reactions of the teachers demonstrated that in general, the developed system is applicable for using in classes and the programming language developed is simple and intuitive. Also, the teachers with whom the tests were performed said that they would use the developed system in their classes.

The tests performed with students also showed positive results. All students demonstrated being enthusiastic and interested in playing with Cozmo to learn with the educational functions. Also, with a brief explanation, the students could easily understand how to interact with Cozmo.

In general, all of the proposed objectives were successfully achieved. Most of the performed tests presented satisfactory results. The teachers observed a great potential on the developed system as an educational tool.

In the future, we plan on improving the web interface to facilitate the interaction with the teachers. We also plan on using Cozmo as a tool on lab sessions of a Robotics course for an Engineering program.

Acknowledgment

Special thanks to FAPESP (São Paulo Research Foundation, Brazil) for the financial support and Hanze University of Applied Sciences for the project development collaboration at the Institute of Engineering. The authors also thank Lianne de Vries from Emmaschool, Shenzhen from CBS Compass and Veronica Becerra Rincon for their valuable feedback during during the evaluation of this project.

References

[1] R. Manseur, "Development of an undergraduate robotics course," in *Frontiers in Education Conference, 1997. 27th Annual Conference. Teaching and Learning in an Era of Change. Proceedings.*, vol. 2. IEEE, 1997, pp. 610–612.

[2] J. M. Tur and C. F. Pfeiffer, "Mobile robot design in education," *IEEE Robotics & Automation Magazine*, vol. 13, no. 1, pp. 69–75, 2006.

[3] S. F. Alves, H. Ferasoli Filho, R. Pegoraro, M. A. Caldeira, J. M. Rosário, and W. M. Yonezawa, "Educational environment for robotic applications in engineering," in *International Conference on Research and Education in Robotics*. Springer, 2011, pp. 17–28.

[4] F. N. Martins, H. C. Oliveira, and G. F. Oliveira, "Robótica como meio de promoção da interdisciplinaridade no ensino profissionalizante," in *Anais do Workshop de Robótica Educacional*, 2012.

[5] J. M. Conrad, "Stiquito for robotics and embedded systems education," *Computer*, vol. 38, no. 6, pp. 77–81, 2005.

[6] S. F. Alves, H. Ferasoli Filho, R. Pegoraro, M. A. Caldeira, J. M. Rosário, and W. M. Yonezawa, "Proposal of educational environments with mobile robots," in *Robotics, Automation and Mechatronics (RAM), 2011 IEEE Conference on*. IEEE, 2011, pp. 264–269.

[7] D. J. Ahlgren, "Meeting educational objectives and outcomes through robotics education," in *Automation Congress, 2002 Proceedings of the 5th Biannual World*, vol. 14. IEEE, 2002, pp. 395–404.

[8] "Study john dewey on education: Impact and theory," https://study.com/academy/lesson/john-dewey-on-education-impact-theory.html/, accessed: 19-09-2018.

[9] F. N. Martins, I. S. Gomes, and C. R. Santos, "Rosos-a free and open-source robot soccer simulator for educational robotics," in *Robotics*. Springer, 2016, pp. 87–102.

[10] F. N. Martins, I. S. Gomes, and C. R. F. Santos, "Junior soccer simulation: providing all primary and secondary students access to educational robotics," in *Robotics Symposium (LARS) and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR), 2015 12th Latin American*. IEEE, 2015, pp. 61–66.

[11] J. Han, "Emerging technologies: Robot assisted language learning," *Language Learning & Technology*, vol. 16, no. 3, pp. 1–9, 2012.

[12] G.-M. Jeong, C.-W. Park, S. You, and S.-H. Ji, "A study on the education assistant system using smartphones and service robots for children," *International Journal of Advanced Robotic Systems*, vol. 11, no. 4, p. 71, 2014.

[13] R. V. Aroca, "Plataforma robótica de baixíssimo custo para robótica educacional," 2012.

[14] C. A. d. S. Reis, H. R. Sarmento, and V. Zaramella, "Ferramenta de auxílio ao desenvolvimento do pensamento computacional: uma plataforma robótica controlada por smartphone," B.S. thesis, Universidade Tecnológica Federal do Paraná, 2014.

[15] "Anki developer home," https://developer.anki.com/, accessed: 02-07-2018.

[16] "Anki installation windows," http://cozmosdk.anki.com/docs/install-windows.html, accessed: 02-07-2018.

[17] "Google Cloud text-to-speech," https://cloud.google.com/speech-to-text/, accessed: 16-07-2018.

[18] "Google Cloud vision api," https://cloud.google.com/vision/, accessed: 16-07-2018.

[19] F. N. Martins, M. Sarcinelli-Filho, and R. Carelli, "A velocity-based dynamic model and its properties for differential drive mobile robots," *Journal of Intelligent & Robotic Systems*, vol. 85, no. 2, pp. 277–292, 2017.

[20] Wikipedia, "Flask," https://pt.wikipedia.org/wiki/Flask_(framework_web), accessed on 16-02-2018.

[21] B. Escola, "A importância da geometria nas séries iniciais," http://educador.brasilescola.uol.com.br/estrategias-ensino/a-importancia-geometria-nas-series-iniciais.htm, accessed on 16-02-2018.