Revista Romana de Interactiune Om-Calculator 10 (3) 2017, 231-249

Educational Mobile Application using Sphero SPRK+ in an Augmented Reality scenario

Teodor Stefanut, Andreea Camelia Tripon

Technical University Cluj-Napoca
26 G. Barițiu street, Cluj-Napoca, Romania
E-mail: teodor.stefanut@cs.utcluj.ro, camelia.tripon1@gmail.com

Abstract. Teaching kindergarten children is a challenging task that requires patience, imagination and very well designed didactic materials that can capture the attention of children and convey knowledge at the same time. Appropriate to the age of pupils, specific games have been developed and integrated into teaching methodologies, in order to help children in understanding and exercising different simple or more complex notions and skills. Complementing traditional teaching methods, technology is currently more present than ever in the everyday activities of children, enabling the implementation of complex learning scenarios, adaptive didactic materials, customized learning paths and optimized integration of gaming with the need of exercising different skills. Some of the challenges met in creating such tools are discussed in this article, showing as a case study that new learning methods and resources can be developed by bridging the virtual world of the mobile-based games with the real one, through the use of Augmented Reality techniques and Sphero SPRK+ robot.

Keywords: Augmented Reality; Sphero SPRK+; Education; Mobile applications.

1. Introduction

Classical educational methods for teaching kindergarten children are mostly based on activities like drawing, singing, playing group games or solving simple tasks, under the supervision and with the guidance of an adult. However, in this educational approach the training and skills exercising is mostly performed only a few hours each day, the responsibility for further practice being passed to the parents who, in the vast majority of situations, do not have the required knowledge to continue adequately these activities.

Kindergarten children have a broad capacity to accumulate knowledge, but they also lose their focus very quickly. As Seefeldt & Wasik (2005) concluded in their study, at age 3 to 5 children can focus about 4-5 minutes on an assigned task and around 10 minutes on activities that present something new and engaging throughout. Teaching methods used need therefore to provide an attractive learning environment, which is able to keep them interested while explaining new notions to pupils and encouraging them to exercise and apply newly acquired skills.

On of the most efficient approach in creating such interactive environments is based on technology, which can bring many benefits to the learning process, making it both fun and interesting. For example, children can listen to an alphabet song, watch movies or cartoons, use tablets or smartphones to play educational games. Technology acceptance is currently not an obstacle for the integration of this approach, as argued by (Flewitt et al, 2015) and supported by the many initiatives coming to life in the form of: interactive whiteboards (Moss et al., 2007; Twiner et al., 2010; Warwick et al., 2010), digital games (Apperley and Walsh, 2012), digital texts (Burnett and Merchant, 2012; Thoermer and Williams, 2012), social interactions powered by media (Carrington and Robinson, 2009) and other new technologies (Wohlwend, 2009).

Augmented reality (AR) allows users to view virtual objects on top of the physical world observed through the mobile device's camera. In education, AR brings numerous advantages and there are infinite ways of using it. It can make every classical learning activity way more interesting. For example, instead of hearing stories and viewing pictures of wild animals, the child can observe them in 3D with proper sounds and animations. Math can become funnier if, for example, a card with a number is augmented with the same number of animals as in the game Math Alive (Alive Studios, 2017).

With the increased development of technology, children currently have easy access to toys like tele-guided cars or robots, which provide higher levels of interactivity than classical ones. Paired with approaches based on augmented reality these gadgets provide support for developing numerous scenarios that are interactive and integrate virtual elements with real ones, from children's surrounding. Based on these technologies we will describe in the following chapters an augmented reality game in which the Sphero SPRK+ robot is a bridge between the real world and a virtual one, designed towards learning purposes.

The natural nature of interactions between children and robots as toys, as well as the benefits in behavioral and communication development have been also explored. In (Faria et al., 2016) the authors describe experiments designed to determine the expressiveness of a simple robot (Sphero in this case) in communication intentions in different contexts, and in inviting people to play and interact with it. (Golestan et al., 2017) discuss the capability of simple robots (Sphero as test case) to help in rehabilitation of children with autism, by engaging them in exercising social and communication skills.

2. RELATED WORK

The use of Augmented Reality techniques in the development of teaching materials has been inspired from traditional learning activities, making them more interactive and attractive for children.

For example, Quiver Education (QuiverVision, 2017) is a game which uses augmented reality to create living drawings. It is designed for children to explore their creative aptitudes. The game consists in printing and coloring the drawings and then using the app to see the drawing coming to life. The interesting part is that the colors used in the real drawing are also used for the augmentation, giving the artist a special sense of ownership. Also the characters are animated and the user can interact with them by tapping on the screen and selecting different actions. The purpose of this game is to make coloring more fun, interesting and to allow the child to interact with the drawings.

Zoo-AR (Zoo-AR, 2017) is another example of educational game based on augmented reality, where children can use custom pictures of animals to display a 3D interactive representation of them. The application provides fun ways to learn about animals' behavior, about how they sound or their size compared to real-life objects. The application involves it's users even more allowing them to take pictures of these animals in different real-life contexts, making learning fun and engaging.

In order to intertwine even more reality and information presented on the screen or learned through the apps, for an even more engaging experience, different methods to integrate remotely controlled robots in learning scenarios have been developed. One of these examples is Sphero Edu (Sphero, 2017) which was designed by the creators of Sphero SPRK+. The application contains a full set of instructions to control the robot and read its state. It also has a feature that allows users to develop small programs that can be executed by the robot. These programs can help users understand basic programming principles as conditional blocks or loops.

Ioannou & Bratitsis, 2017, propose an application dedicated to kindergarten children, which uses Sphero SPRK to teach the notion of speed.

Another example where Sphero is used as an educational mean is presented in article (Rodriguez Corral et al, 2016). Since Sphero offers free SDKs to allow developers design their own programs, the authors are using this particularity to teach students basic Object Orientated Programming (OOP) principles. The paper presents a study where university students, with no background in programming, can learn basic C# and programming principles by designing a program which controls the Sphero robot. The students claim that this learning method is more interesting and effective because they can see the practical use of programming by controlling Sphero's moves.

SMAUG is a collaborative AR game that also uses Sphero as the main character. The game is presented in article (Panzariu & Iftene, 2016), where the authors show that Sphero's control can be shared to complete more complex tasks. The environment is augmented with objects like diamonds, dragons or treasures and Sphero has to collect or avoid these virtual objects to complete the level's goal.

Sphero and AR meet in another application made by the creators of Sphero: The Rolling Dead (Sphero, 2013). The gameplay consists in using Sphero as the main character in order to defeat the zombies that are spawning in an augmented environment.

Similar to these applications, our proposal combines the possibilities of Augmented Reality applications with the engagement of remotely controlling the robot. However, the purpose of the application is not entirely for amusement. While engaging and fun, the activities have been designed with the educational purpose in focus, aiming at teaching children useful notions about numbers, animals and geometrical figures while enabling them to exercise fine movement skills and coordination. Furthermore, to tease the curiosity of young minds, we have used in our experiments the Sphero SPRK+ version, which comes with a transparent shell that allows the observation of mechanical parts of the robot.

3. SPHERO SPRK+

Sphero SPRK+ is a ball-shaped robot designed for educational purposes which can be controlled by a mobile device or computer. The robot can move in any direction with the speed indicated by the user and has 2 RGB LEDs, feature which makes this toy even more attractive for kids. One important design feature is that Sphero's shell is transparent and kids can actually see its electronics (see Figure 1).

Hardware specifications

The robot is equipped with accelerometer and gyroscope sensors, which are useful to determine collisions, measure velocity or determine robot's orientation.

The communication between Sphero and the controlling device is made through Bluetooth. Sphero SPRK+ uses Bluetooth 4.0, known as Bluetooth Smart or BLE. This feature is the main upgrade from the previous version, Sphero SPRK, which required additional steps in order to establish a connection. The robot can be controlled from a distance up to 30 m.

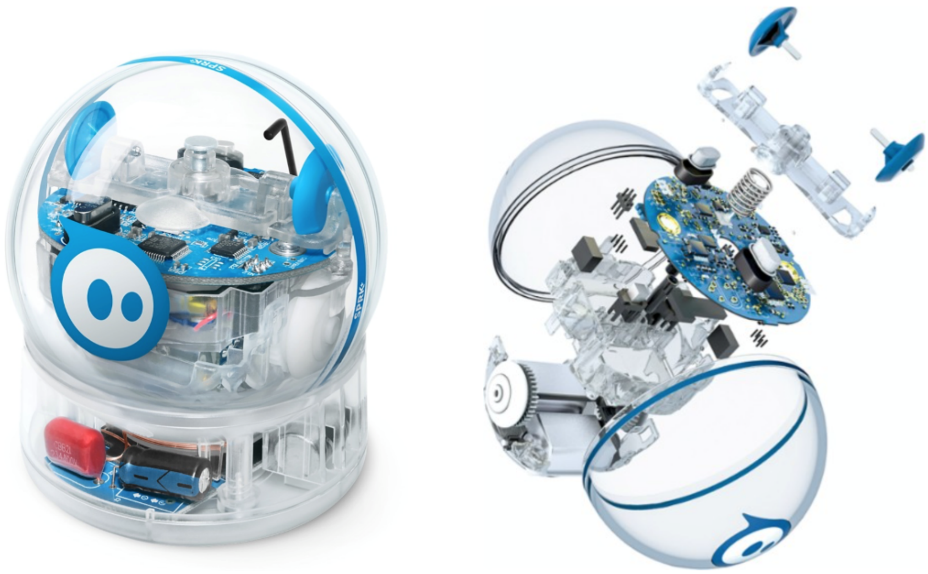

Figure 6. Sphero SPRK+ components (Sphero, 2017)

Software specifications

Sphero's creators offer support for different platforms as Android, iOS, Windows, Unity, through a Software Development Kit (SDK). These libraries are used to send proper commands to the robot. Sphero understands commands such as: driving in a direction with a selected speed, changing the color of the LEDs or changing the orientation. The SDK also handles connection lifecycle and can receive asynchronous information from the sensors.

3. IMPLEMENTATION CONSIDERATIONS

In order to include Augmented Reality capabilities into our application we have considered Vuforia (PTC Inc., 2017), which according to the official website is currently "the world's most widely deployed AR platform". The main feature that recommended Vuforia in our case is the extended capabilities in tracking predefined and custom-defined markers, which we attempted to use for tracking Sphero's position relative to the mobile device.

3.1 Sphero Tracking Algorithm

In order to be able to immerse the real Sphero robot into the virtual world presented on the device's screen, it is absolutely necessary to always know its position in the 3D scene. This position will change during the game based on two main factors, which both reside in the real world:

- movement of the robot according to user's commands
- movement of the device when the user changes position

This makes it impossible to reliably establish the position of the robot in the 3D virtual world based on some initial setting at the start of the activity and mandates the real-time tracking of the position throughout the game. In the attempt to meet these requirement, we have tested two different tracking approaches: using Vuforia library and using a custom algorithm to detect light sources.

Tracking Sphero with Vuforia

As already mentioned, Vuforia provides out-of-the-box support for tracking predefined and custom-defined markers. When comes to 3D objects, the tracking mechanism is setup by scanning (taking photos of) all the faces of the object and creating a detailed map of feature points (that can be uniquely identified by Vuforia from the images). Map construction is automatically performed when pictures taken are uploaded into the Vuforia developer portal. The tracking algorithm is relying on matching feature points detected at scanning time on the object with the ones recognized and tracked from the video feed at runtime.

One of the first setbacks we encountered was related to the fact that Vuforia can track objects that have many edges and contrasting areas, which is exactly the opposite of Sphero’s shape and colors: no edges and only shiny surfaces (see Figure 2, left). As a result, the feature points map obtained was too general (too few details available) to enable a reliable detection of Sphero.

Attempting to improve the number of details available on the object, we decided to wrap the ball with a material that has a lot of features. A picture of the used fabric was first uploaded to Vuforia developer portal and it was evaluated with 5 stars rating. We can see the wrapped Sphero and the coverage from scanning in Figure 2 (right). As it is observed in the picture, Vuforia found 1148 points of interest that will be used to track the ball.

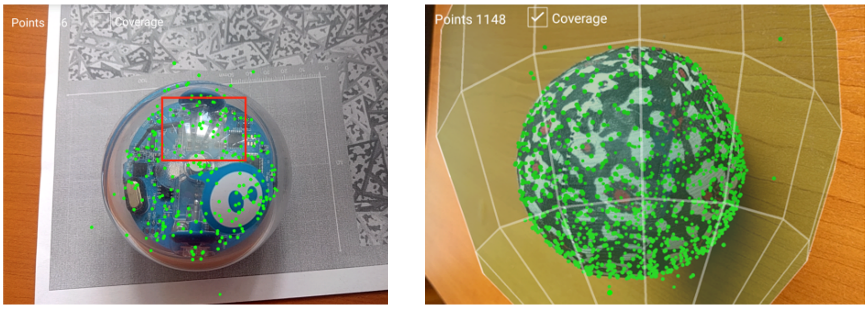

Figure 7. (left) Sphero tracking points. (right) Wrapped Sphero and tracking points

Although the scanning algorithm found many points of interest, the tracking results were far from satisfactory. The ball was detected only from small distances and if the room was well-lit. The tracking was quite unreliable if the ball was moving (especially at higher speeds) or the distance was more than 50 cm.

While still trying to rely on the embedded Vuforia tracker, we have experimented with speed limitations, attempting to significantly improve overall tracking. However, the maximum reliable detection distance could not be consistently improved, 50 cm being far less than the necessary useful distance required (keeping in mind that the user will probably stand when using the app and could move around the room).

Because the results from this approach were not good enough, we decided to use another method for tracking.

Tracking Sphero as a Light Source

Since Sphero's characteristics were not compatible with Vuforia's embedded tracker, we decided to attempt a custom tracking approach based on the light source and the spherical form of the robot. Using the image captured from the video feed, in each frame, the following algorithm is applied:

- transform image to grayscale
- binarize image using a threshold of 245
- noise reduction by applying the morphological operation "opening" with a 7x7 kernel
- find the contours from the remaining data in the image. The center of the contour is sent to Unity, where an algorithm is applied to transform the 2D point in a 3D object.

These operations are written in C++ and use the OpenCV library to process the image in real time, directly on the mobile device. The result of this algorithm is shown in Figure 3.a

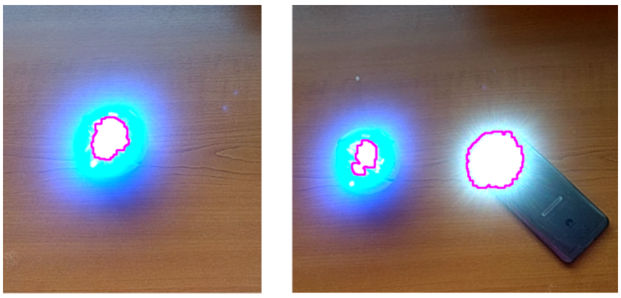

Figure 8. Processed image with OpenCV

The main advantages of the propose approach are: (a) independence of the light's color (processing is performed in grayscale); (b) processing speed of about 50ms for a 800x480px image on a Nexus 5x device, which enables real-time tracking; (c) experimentally tested reliability in tracking the robot, even at higher speed and further distances.

The disadvantage of this method comes when more light sources with similar light intensity can be identified in the image (see Figure 3b.)

3.2 Transform 2D point to 3D object

Successfully tracking Sphero in the real world solves only part of our endeavor of integrating the robot into our virtual environment, and enabling it to interact with the virtual objects.

When the tracking algorithm has finished processing, we obtain the 2D screen coordinates of the center of Sphero. From this point onward, we could continue our collision detection analysis in 2D screen space, by using the projection of the 3D objects from the virtual world. Although simpler, and suitable for some situations, this approach has a major drawback: without any information on the depth of Sphero's position into the virtual scene, false collisions will be reported when the robot is actually behind a 3D virtual object.

Consequently, correct collision detection can be performed only in the 3D virtual space and only if Sphero can be represented as a virtual object in the scene. In order to fulfill this requirement, the next necessary step is to translate the 2D screen coordinated of the robot into 3D scene coordinates, which involves identifying the depth value.

The virtual scene from the device is projected with the use of a specific marker, detected by Vuforia, which provides information on 3D world orientation, distance from the device to the marker etc. According to the instructions of use, this marker needs to be placed on the floor, which represents the same surface on which our Sphero robot moves. Consequently, the plane determined by the marker in the virtual world is actually the virtual representation of the floor from the real world.

Based on this assumption, we can determine Sphero's 3D position by intersecting the ray defined by the virtual camera and the screen point with the plane determined by the marker as observed in Figure 4.

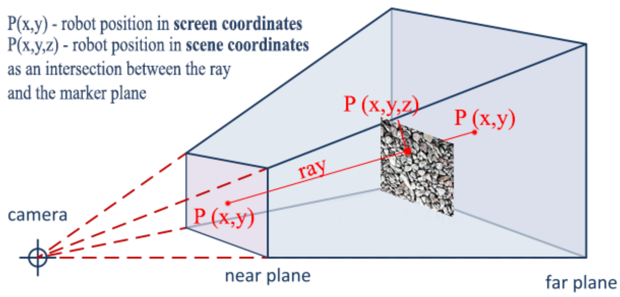

Figure 9. Screen position to 3D position

After knowing the 3D position of Sphero's center, all we have to do is create a 3D object with Sphero's dimensions and move it to the computed point in scene. As this is a 3D operation, the avatar of the robot will be automatically scaled based on the determined depth, so collisions with the other virtual objects will be more accurate.

The algorithm performs well on medium performance devices and the position is updated in each frame following the light source. While displaying the virtual avatar of Sphero can represent a second confirmation of robot's position, is not required in order to perform collisions processing. Hiding it can further improve the sensation of intertwining between the real world and the virtual one.

3.3 Sphero Auto-Calibration Algorithm

While playing the game, children will control Sphero using the mobile device. Depending on the robot's initial orientation, and later into the game on the user's position changes, **forward direction** perceived by the user might not be the same as the one perceived by Sphero. Although children might be able to infer the direction of movement for familiar looking objects (ex. toy car) and to correctly assess and correct a wrong "move forward" command, it is not the case with Sphero, being a spherical object.

As a result, in order to synchronize the **forward direction** of the robot with the one perceived by the user, a calibration process is required. In other words, we should teach the robot which way is forward for us. This step consists in aligning the robot backlight in front of the user, by rotating the robot.

Although calibration is not difficult and children could easily learn to perform it, they would prefer to just dive into the game, without any additional steps for the setup. Also, they can do mistakes in calibration which will result in sending wrong directions to Sphero, and frustration for them. For these reasons, we decided to develop an auto-calibration feature. This consists in automatically aligning the robot to user's forward direction, when the game is started.

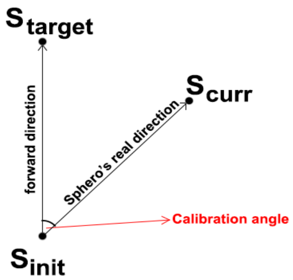

Figure 10. Calibration algorithm representation

The algorithm is represented in Figure 5 and has the following steps:

- choose a random position S_{target} considered to be in front of Sphero based on camera's position; we mark the distance between Sphero and position S_{target} with **d**
- move Sphero with the direction "forward", as understood by the robot (0 degrees rotation), until **d** is the distance between Sphero's initial position S_{init} and Sphero's current position S_{curr}
- compute the angle between the vectors determined by $S_{init}S_{target}$ and $S_{init}S_{curr}$ and adjust Sphero's rotation with this angle.

3.4 Software Dependecies

Sphero Android SDK (Sphero SDK, 2017) – a library which is used to send commands to the robot. Since Bluetooth 4.0 is a newer version, the open source SDK is only in beta version and is not available for Unity. To overcome this obstacle, we created an Android plugin for Unity which acts as a bridge between Unity procedures and Java methods provided by the Sphero SDK. The plugin supports simple instructions like connecting to the robot, sending commands to move or to change colors and calibrating.

OpenCV (OpenCV Team, 2017) – is a library for image processing. It is used to detect the Sphero object in the image captured by the camera. Because it is not available for Unity, we wrote a small program in C++ which is then compiled in a shared object (.so) and called using NDK directives. The shared object library has dependencies to the open source OpenCV library and contains the processing method necessary to detect the robot.

Vuforia – the library we have chosen for AR support. It has versions for Android, iOS or Unity. It is used directly as a plugin for Unity and provides scripts and prefabs (objects already built in the library that include the necessary properties and scripts) to recognize markers and show augmented objects on top of the target area. The markers supported by Vuforia are: image, frame, object (cylinder or custom object that is scanned and loaded into the app). For this application, we used a simple image target evaluated with 5 stars on Vuforia website. All the scenes that use augmented reality use a prefab called ARCamera and an ImageTarget.

4. GAME STORIES

As it we previously mentioned, the game is divided in five levels with progressive difficulty. Each level will teach the child one different concept appropriate to his age (numbers, geometrical figures, animals, etc). The user can play a level, only if he completed all the previous levels.

4.1 Common Elements in Levels

A level is represented as a Unity scene. Sphero is represented in all scenes by a translucent sphere and its position is updated according to the light source center. The collision detection is performed in every frame and is based on the virtual representation of Sphero and the virtual objects placed on the scene. A collision between virtual Sphero and another object in scene is followed by visual and audio indicators: Sphero will have green lights when the object is correct and red lights when a wrong object is selected and a proper sound will be played in both cases.

4.2 Star rating

To evaluate a level, we use the number of mistakes and the time spent to achieve the level's goal. Each level has different thresholds for these parameters, depending on the number of elements and the complexity of the scene. The performance on each level will be ranked with a rating from 1 to 3 stars. Three stars rating can be achieved only if the user did not make any mistakes like selecting a wrong object and if he/she completed the level in a reasonable time.

4.4. Levels Description

Level 1 – Colors and 3D shapes

This level will help children to accumulate basic knowledge about colors and shapes. Although this kind of exercise is commonly used in kindergartens, this method is funnier and more interesting. The scene consists of 7 geometrical figures like cube, sphere, cylinder which are colored in 4 base colors (see Figure 6). The gameplay consists in selecting the correct objects as requested in a message at level start and after each correctly selected object. There are no restrictions regarding the path to the correct object, but Sphero should not collide with a different object because it will be considered a mistake.

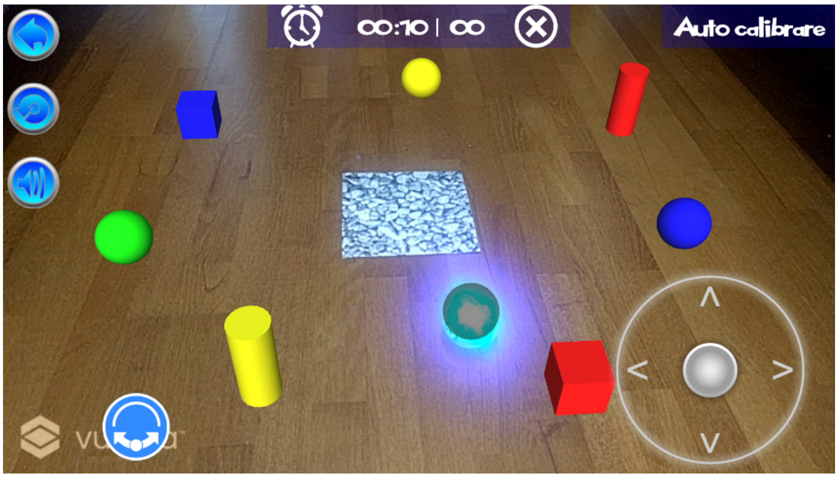

Figure 6. Level 1 - Colors and 3D shapes

Level 2 – Learning numbers

Since numbers are one of the most important concepts that should be learned by children, the second level proposes a fun way to learn counting. The scene shows the numbers from 1 to 9 in a random order (see Figure 7) and the child must select all numbers in ascending order. If mistakes are made, these will affect the final score, but the user can continue selecting another number without restarting the level.

Figure 7. Level 2 - Learning numbers

Level 3 – Wild Animals

One important part in child's development at a preschool age is to learn the animals and their habitat. Because some of these animals could be dangerous, this application proposes another way than seeing them in real life. 3D projections of animals will appear in the scene (see Figure 8) and the child can interact with them by tapping the screen. For now, the interaction is basic: play the sound of the animal and execute an animation, but more complex interactions will be added.

To pass this level, user has to select each animal in the correct order. The order will be displayed step by step: first message is shown at level start and the following are displayed after each correct selection.

Figure 8. Level 3 - Wild animals

Level 4 – Drawing dot-by-dot

Drawing is another common activity in kindergarten. For this application, we included the dot-by-dot drawing technique which is both fun and educational. Dot-by-dot drawing implies showing a pattern and the control points of that drawing. The child should connect the points in the correct order to obtain the final figure. For this application, we included 3 simple patterns: house, flower, table. The control points are shown as letters (see Figure 9). Therefore, while playing this level, the child will learn the first letters of the alphabet and to draw simple figures.

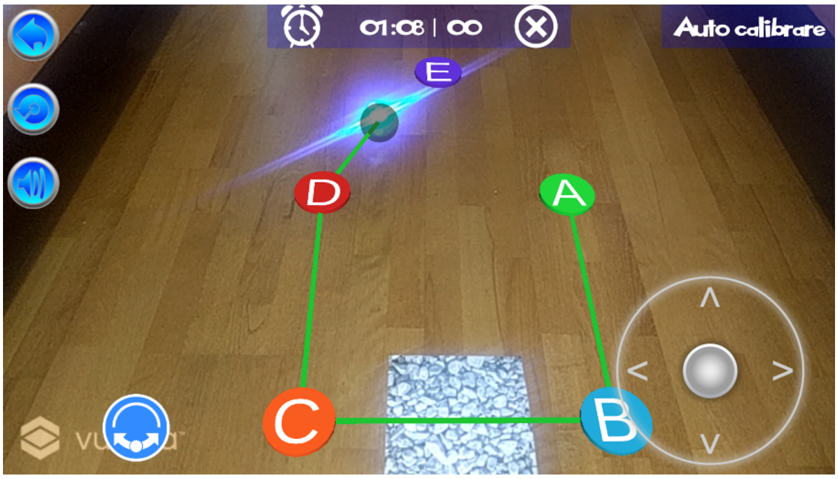

Figure 9. Level 4 - Drawing dot-by-dot

Level 5 – Collect Chests

This level is a little different than the others. It does not teach the child any new concepts but rather it tests his abilities to control and to think ahead his moves.

The scene is composed of 3 moving obstacles and 3 chests (see Figure 10). To complete the level, the child must collect all the chests but without touching the obstacles. The first two obstacles are boxes that are moving with different speed on Ox axis. To collect the first two chests, user should observe the course and the speed of the boxes and carefully control Sphero to avoid those obstacles. The third obstacle is a moving rock that is rotating and protecting the final chest. The child must collect the final chest when the rock passed, but he should be careful not to get hit when it rotates back.

When Sphero is hit by one of the obstacles, the level is reset and Sphero is sent back to the starting point. No mistakes are permitted to reach level's goal.

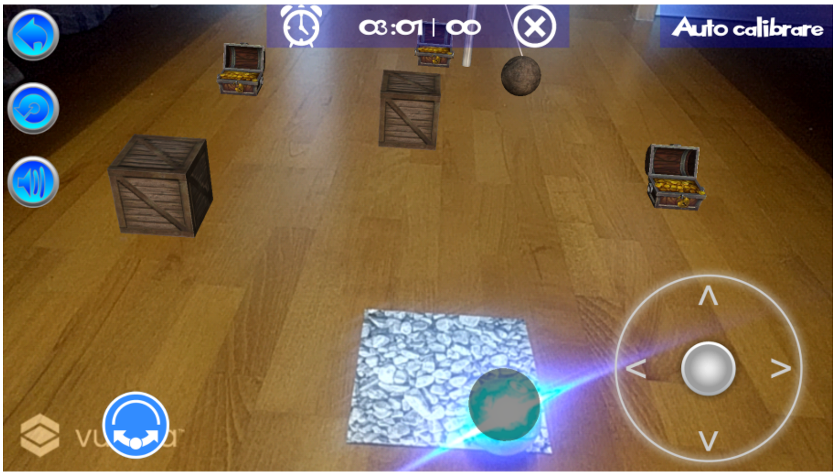

Figure 10. Level 5 - Collect chests

5. Conclusion

Current article proposes an approach for the development of interactive teaching applications that intertwine Augmented Reality techniques with the engaging interaction with remotely controlled robots. Proposed application can be used both at home or in kindergartens and it does not require a supervisor to explain the tasks to the child. The messages are clear and always appear in two forms: written and audio, making the communication process easier.

The actual learning process happens when the kid controls the robot into achieving level's goal. If the child completes a level, it is considered that he accumulated the necessary information from that level and can pass to the next one. The Achiever typology from games theory is used at this point with two types of reward: (1) accessing the next level; (2) number of stars received at the end of the current level.

The star rating given at level end acts like a reward and a grade for finishing the level. To not disappoint kids, at least one star is given just for finishing the level. This rating will stimulate the player to try again and receive the maximum number of stars. This means that a small rating will most likely make the child replay the level in order to get better rating while also involuntarily repeating the notions learned. A three-star rating means that the concepts are learned and understood correctly, without mistakes and in a good response time.

As a pilot application, the current version includes only a limited set of concepts, that can be easily extended in the future. At the same time, testing and validating the effects of this approach in real-case studies, involving kindergarten children and educators, is one of our main future objectives.

References

Alive Studios (2017). Math Alive, https://alivestudiosco.com/ [Online]

Apperley, T. and Walsh, C. (2012) What digital games and literacy have in common: a heuristic for understanding pupils' gaming literacy. *Literacy*, 46(3) pp. 115-122

Burnett C. and Merchant G. (2012) Learning, Literacies and New Technologies: The Current Context and Future. *In J. Larson and J. Marsh (Eds.) The SAGE Handbook of Early Childhood Literacy (2nd edtn)*, London: Sage, pp. 575-586.

Carrington, V. and Robinson, M. (Eds.) (2009) Digital Literacies: Social Learning and Classroom Practices. *Sage Publications Inc.* ISBN-13: 978-1-84787-037-7

Faria, M., Costigliola, A., Alves-Oliveira, P., Paiva, A. (2016) Follow me: Communicating intentions with a spherical robot, *25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, New York, NY, pp. 664-669

Flewitt, R., Messer, D., Kucirkova, N. (2015). New directions for early literacy in a digital age: the iPad. *Journal of Early Childhood Literacy*, 15(3) pp. 289–310.

Golestan, S., Soleiman, P., Moradi, H. (2017) Feasibility of using Sphero in rehabilitation of children with autism in social and communication skills, *International Conference on Rehabilitation Robotics (ICORR)*, London, pp. 989-994.

Ioannou, M., Bratitsis, T. (2017) Teaching the Notion of Speed in Kindergarten Using the Sphero SPRK Robot, *IEEE 17th International Conference on Advanced Learning Technologies (ICALT)*, Timișoara, pp. 311-312.

Moss, G., Jewitt, C., Levacic, R., Armstrong, V., Cardini, A. & Castle, F. (2007) The interactive whiteboards, pedagogy and pupil performance evaluation. Research Report 816, London: DfES.

OpenCV Team (2017), Open Source Computer Vision Library, https:// opencv.org/ [Online]

Panzariu, M.N., Iftene, A. (2016) Sphero - Multiplayer Augmented Game (SMAUG), *Proceedings of International Conference on Human-Computer Interaction - RoCHI 2016*, Iași, pp. 46-49. ISSN 2501-9422

PTC Inc. (2017) Vuforia, https://www.vuforia.com/ [Online]

QuiverVision (2017). Quiver Education, http://www.quivervision.com/apps/quiver-education/ [Online]

Rodriguez Corral, J. M., Morgado-Estevez, A., Molina Cabrena, D., Perez-Pena, F., Amaya Rodriguez, C. A., Civit Balcells, A. (2016) Application of Robot Programming to the

Teaching of Object-Oriented Computer Languages, *International Journal of Engineering Education*, 32 (4) pp. 1823-1832

Seefeldt, C., Wasik, B.A. (2005) Early Education: Three, Four, and Five Year Olds Go to School (2nd Edition). *Pearson; 2 edition (June 30, 2005)*. ISBN-13: 978-0131190801

Sphero (2017). Sphero Edu, https://edu.sphero.com/ [Online]

Sphero (2017). Sphero Android SDK, https://github.com/orbotix/Sphero-Android-SDK [Online]

Sphero (2017). Sphero SPRK+, http://www.sphero.com/sprk-plus [Online]

Sphero (2013), The Rolling Dead: http://blog.sphero.com/blog/spheros-the-rolling-dead-now-available-on-ios-and-android/ [Online]

Thoermer, A. and Williams, L. (2012), Using Digital texts to promote fluent reading *The Reading Teacher*, 65 (7) pp. 441–445

Twiner, A., Coffin, C., Littleton, K. and Whitelock, D. (2010) Multimodality, orchestration and participation in the context of classroom use of the interactive whiteboard: a discussion. *Technology, Pedagogy and Education*, 19 (2) pp. 211-223.

Warwick, P., Mercer, N., Kershner, R., & Kleine Staarman, J. (2010). In the mind and in the technology: The vicarious presence of the teacher in pupils' learning of science in collaborative group activity at the interactive whiteboard, *Computers and Education*, 55 (1) pp. 350–362.

Wohlwend, K. (2009) Early adopters: Playing new literacies and pretending new technologies in print-centric classrooms, *Journal of Early Childhood Literacy* 9(2) pp. 117–140

Zoo-AR (2017). Zoo-AR App, http://zoo-ar.com [Online].