Original software publication

Software architecture for YOLO, a creativity-stimulating robot

Patrícia Alves-Oliveira [a,b,c,*], Samuel Gomes [c,d], Ankita Chandak [e], Patrícia Arriaga [a], Guy Hoffman [b], Ana Paiva [c,d]

[a] Iscte-Instituto Universitário de Lisboa, CIS-IUL, Lisbon, Portugal
[b] Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY, USA
[c] INESC-ID, Lisbon, Portugal
[d] Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal
[e] Computer Science Engineering, Cornell University, Ithaca, NY, USA

ARTICLE INFO

ABSTRACT

YOLO is a social robot designed and developed to stimulate creativity in children through storytelling activities. Children use it as a character in their stories. This article details the artificial intelligence software developed for YOLO. The implemented software schedules through several Creativity Behaviors to find the ones that stimulate creativity more effectively. YOLO can choose between convergent and divergent thinking techniques, two important processes of creative thought. These techniques were developed based on the psychological theories of creativity development and on research from creativity experts who work with children. Besides promoting creativity, this software allows the creation of Social Behaviors that enable the robot to behave as a believable character. We built 3 main social behavior parameters: Exuberant, Aloof, and Harmonious. These behaviors are meant to ease immersive play and the process of character creation. The 3 social behaviors were based on psychological theories of personality and developed using children's input during co-design studies. Overall, this work presents the design, development, and usage of social robots that might nurture intrinsic human abilities, such as the ability to be creative.

Code metadata

Current code version	v0.4
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX_2019_242
Legal Code License	CC Attribution 4.0 International
Code versioning system used	git
Software code languages, tools, and services used	Python
Compilation requirements, operating environments & dependencies	Raspbian Stretch Lite OS
If available Link to developer documentation/manual	https://github.com/patricialvesoliveira/YOLO-Software/wiki
Support email for questions	Samuel Gomes: samuel.gomes@tecnico.ulisboa.pt and Patrícia Alves-Oliveira: patricia_alves_oliveira@iscte-iul.pt

Software metadata

Current software version	v0.4
Permanent link to executables of this version	https://github.com/patricialvesoliveira/YOLO-Software
Legal Software License	CC Attribution 4.0 International
Computing platform / Operating System	Linux (Raspbian Stretch Lite OS)
Installation requirements & dependencies	Raspbian Stretch Lite OS
If available link to user manual	https://github.com/patricialvesoliveira/YOLO-Software/wiki
Support email for questions	Samuel Gomes: samuel.gomes@tecnico.ulisboa.pt and Patrícia Alves-Oliveira: patricia_alves_oliveira@iscte-iul.pt

* Corresponding author.
E-mail address: patricia_alves_oliveira@iscte.pt (P. Alves-Oliveira).
URL: http://www.patricialvesoliveira.com (P. Alves-Oliveira).

Fig. 1. YOLO, a social robot that can boost creativity in children.

1. Introduction

Creativity is one of the most sought-after skills, with recognized benefits in education, mental health, and professional success [1,2]. It is associated with states of joy, play, efficiency, and pleasure [3,4]. When stimulated in childhood, it promotes overall development with specific benefits in learning and adaptation. [5]. Creativity has been stimulated through the use of different intervention programs with promising effectiveness levels, demonstrating the potential to develop creativity with proper training [6,7]. However, most of these interventions developed for children lack elements of joy and fun. Instead, they are considered similar to test-like exercises that can hinder their creative expression. The fast pace of technology development enabled the design of new tools for exploring the context of creativity stimulation [8] (e.g., the CUBUS virtual environment for storytelling with emotionally evocative characters [9]).

In our work, we aim to expand the range of technologies used for creativity stimulation by incorporating social robots to catalyze this ability. With this in mind, we designed and developed YOLO (Your Own Living Object), an original social robot to be used as a toy during children's play times (see Fig. 1). YOLO belongs to a new generation of technological toys meant to stimulate creative abilities in children. This robot is envisioned as a character children use during storytelling. By having a small-size and a light-weight design, YOLO can be manipulated by children as if it was a traditional toy (similarly to what they do with dolls or toy cars). The added-value of this robot is that it can increase the creative thought process of children during story creation by providing novel ideas for storylines. Because YOLO is a non-anthropomorphic robot, it interacts with children using alternative but effective interactive modalities. These comprise variable motion and illumination profiles. In this paper, we detail the software behind YOLO. The software is composed of Creativity and Social Behaviors whose design was grounded on creativity research [10], the Big Five personality model [11], and co-design

sessions with children [12]. We release the robot's software[1] along with its installation guide[2] in open access. This software should be exclusively installed on YOLO hardware [13].

2. Background and problems

2.1. Creativity research

As societies develop into creativity-based economies, innovative and creative problem solving and the ability to collaborate are becoming must-have skills [1,14]. However, around the age of 7–9 years old there is a decline in children's creativity abilities known as the "creative crisis" [15]. Research has shown that everyone has the potential to be creative and that creativity can be nurtured if stimulated [16,17]. In our work, we aim to contribute to the increase in children's creativity by using a social robot. It therefore becomes imperative to stimulate this ability at a young age. In particular, emphasizing the school environment where children spend most of their time. While some schools already feature activities (e.g., storytelling) to support the promotion of children's creative thinking [18], current activities can be challenging to integrate into traditional classroom formats as they need preparation time and are not formally included in the school curriculum [19]. Technologies – such as social robots – appear as a more effective tool to apply in these contexts.

2.2. Robots for creativity

Robots have been programmed with a deep variety of socially intelligent behaviors and affective states; thus, allowing them to be perceived as social actors [20,21]. Additionally, due to their physical and interactive nature, they become a technology that can uniquely impact creativity stimulation. Ali, Moroso, and Breazel [22] demonstrated that a robot displaying creative behaviors positively influenced the creativity of children. The authors found that children who interacted with a creative robot generated more ideas, explored more themes, and were more original, than children who interacted with a non-creative robot [22]. Additionally, Gordon et al. [23] demonstrated that children become more curious, an important creativity trait, when interacting with a curious robot. The authors found that these children posed more questions and become avid explores, compared to children who interacted with a non-curious robot [23].

3. Software framework

3.1. Software architecture

The architecture of our software includes several modules that manipulate data at different levels of abstraction from the low-level sensors and actuators to high-level behaviors. Fig. 2 shows the scheme of these modules and how they interact. Each module is explained in the next sections.

3.2. Control

This module has two main functions: first, it extracts data associated with the robots' sensors and translates into a programmable format. Second, it instructs the actuators what to do based on the software calls.

---

[1] Download YOLO Software: https://github.com/patricialvesoliveira/YOLO-Software.

[2] Link to YOLO Software installation guide: https://github.com/patricialvesoli veira/YOLO-Software/wiki.
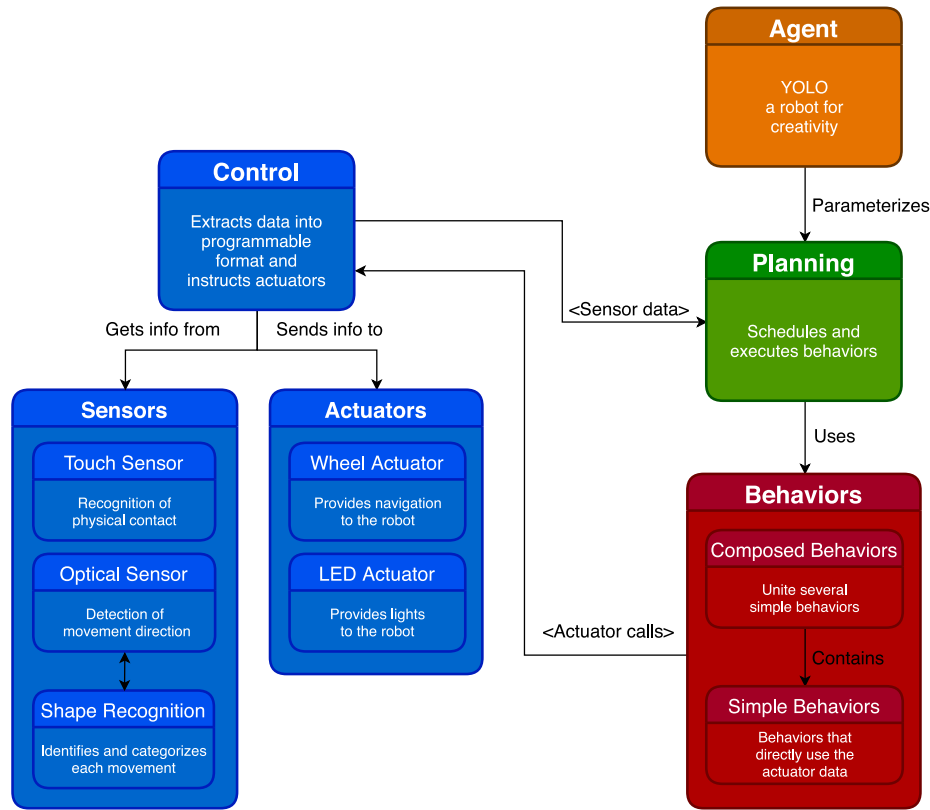
Fig. 2. Architecture of the modules that compose YOLO software.

The *touch sensor* of YOLO indicates to the robot that physical contact was established, and the *optical sensor* observes the differences in position to detect the direction of movement. The sensors are updated at each moment. The *shape recognizer* dynamically identifies and characterizes each movement using Machine Learning (ML). The pre-trained K nearest neighbor (KNN) algorithm determines a shape using the robot motion sensors which capture coordinates in 3 s intervals [24]. Fig. 3 depicts the ML workflow. We trained the model by collecting raw coordinates and converting these coordinates into a feature vector using the convex hull algorithm [25].[3] Every time a movement is detected, KNN is used to determine the closest matching shape from the training data. Simulations with a computer mouse showed us that with n = 3, KNN provided high accuracy (94%). Therefore, we used this parameterization. The current ML model was trained with the physical robot and can recognize with an 80% success rate the following shapes: circle, rectangles, loops, curls, spikes, and a straight line (see Fig. 4).

YOLO actuators include the *Wheel Actuator* and *LED Actuator*. While *Wheel Actuator* receives direction and speed values and moves the wheels' motors accordingly, *LED Actuator* receives a color and brightness level and displays it in the robot's LEDs.

3.3. Behaviors

The *Behaviors* module coordinates the simultaneous execution of different actuators based on given parameters. The intended behavior arises from the simultaneous execution of different actuators. To simplify the development process, we divided behaviors into more concrete *Simple Behaviors*, which directly use the actuator data and *Composed Behaviors*, which unite several simple
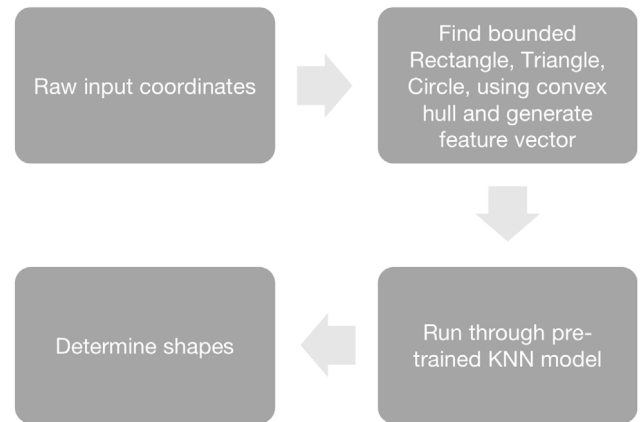

Fig. 3. Workflow of the ML algorithm for shape recognition used by YOLO.

behaviors. *Simple behaviors* directly call the *Control* module. These behaviors consist of assigning different light behaviors (different colors, animations, and brightness) to different movement configurations (different movement patterns at varying speed).[4] *Composed behaviors* can be used to define the social behaviors which YOLO exhibits, such as Exuberant, Aloof, and Harmonious.[5]

---

[3] More details about our shape recognizer algorithm are present at this link: https://github.com/patricialvesoliveira/YOLO-Software/wiki/Algorithm.

[4] Examples of simple behaviors are included with the software documentation: https://github.com/patricialvesoliveira/YOLO-Software/wiki/SimpleBehavior-Hierarchy.

[5] Composed behaviors are further explained at the software documentation: https://github.com/patricialvesoliveira/YOLO-Software/wiki/ComposedBehavior.

Table 1
Software functionalities considering the sensors used, the input collected, the actuators in place, and the output provided.

| Sensor	Input	Actuator	Output
Touch sensor	Ability to recognize when the robot is being touched.	LED lights	The robot displays white lights while being touched, refrains from performing any behavior. When not sensing touch, the robot displays colors associated with its different social behaviors.
Optical sensor	Recognition of the direction of the play patterns of children while manipulating the robot.	Omni wheels	Imitating the collected movement patterns.
Time	Stage of the storytelling that children are currently engaged in.	Omni wheels and LED lights	The robot performs a creativity technique according to the storytelling arc.
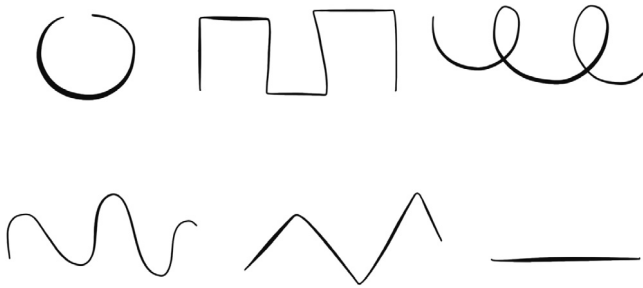
Fig. 4. The lines illustrate the movement shapes performed by YOLO that are recognized by our algorithm. Illustrated shapes are: circle, rects, loops, curls, spikes, and straight line.

3.4. Planning

The *Planning* module schedules the behaviors in each moment of the interaction, executing specific ones based on the current interaction state. In order to trigger new interaction states, *Planning* module uses the data extracted from the sensors which the *Control* module provides. A flowchart illustrating the schedule provided by the *Planning* module is depicted in Fig. 5.

4. Software functionalities

A primary function of this software is to serve as an Application Programming Interface (API) that enables any user the opportunity to design personalized behaviors for YOLO, consequently providing the possibility to generate new behaviors and interaction modes.[6] The robot can receive information from the environment (input) and express different interactive behaviors (output). Table 1 lists pre-sets that were developed for YOLO to act as a social robot that can stimulate creativity in children.

Since each aspect of the robot is controllable and parameterizable, behaviors can be tweaked, created and mixed. To demonstrate the API functionality, we conducted testing sessions in which we asked two participants unfamiliar with YOLO software to create different behaviors for the robot. One of the participants had a background in Computer Science and the other in Psychology. The participants were instructed to choose beloved characters from animation movies and to create a behavior for the robot that would resemble the behavior of those characters. The examples created by the participants were Mickey, Barbie, Bugs Bunny, and Genie from Aladdin.[7]

4.1. Implementation

In our work, we developed a software that gives life to the social robot YOLO, whose interaction is meant to display Creativity and Social Behaviors.

4.2. Creativity behavior

In our specific application scenario, YOLO acts as a character that can trigger new directions in children's stories that otherwise would not emerge. During story creation, a combination of divergent (i.e., broad gathering of multiple ideas) and convergent thinking (i.e., narrowing down possibilities to create a coherent story plot) is required [26–28]. We have chosen two techniques that are often used to stimulate creativity, named "mirroring" and "contrasting" [10] (see Fig. 6)[8]:

- **Contrasting —** This technique is used to stimulate divergent thinking [29]. In the Contrast technique, YOLO provides stimuli unrelated to the storyline that children are exploring at the moment, producing an opportunity to explore new directions in the plot. This leads to heightened action and interesting plot twists in the stories of children.
- **Mirroring —** This technique is used to stimulate convergent thinking [30]. When using the Mirror technique, YOLO provides stimuli that are connected with the storyline that children are exploring, leading to the elaboration and convergence of story ideas. This leads to the emergence of interesting details about a character, a scenario, or an action in the story.

4.3. Storytelling arc

Successful and satisfying stories follow a storytelling arc [31, 32]. According to the Theory of Dramatic Structure, each story has five acts: exposition, raising action, climax, falling action, and dénouement [31,32]. These five acts can be modified and adapted to the dramatic structure of short stories, fables, or fairy-tales. In our software, we considered a short-story format similar to what is used in children's stories [33]. Therefore, we divide the narrative of a story in the following phases:

- **Rising action —** Characters are introduced, a context is given to the story, and the story builds. During this stage, YOLO stimulates convergent thinking by using the mirror creativity technique (see Section 4.2);
- **Climax —** The story reaches the point of greatest tension. During this stage, YOLO stimulates divergent thinking by applying the contrast creativity technique (see Section 4.2);
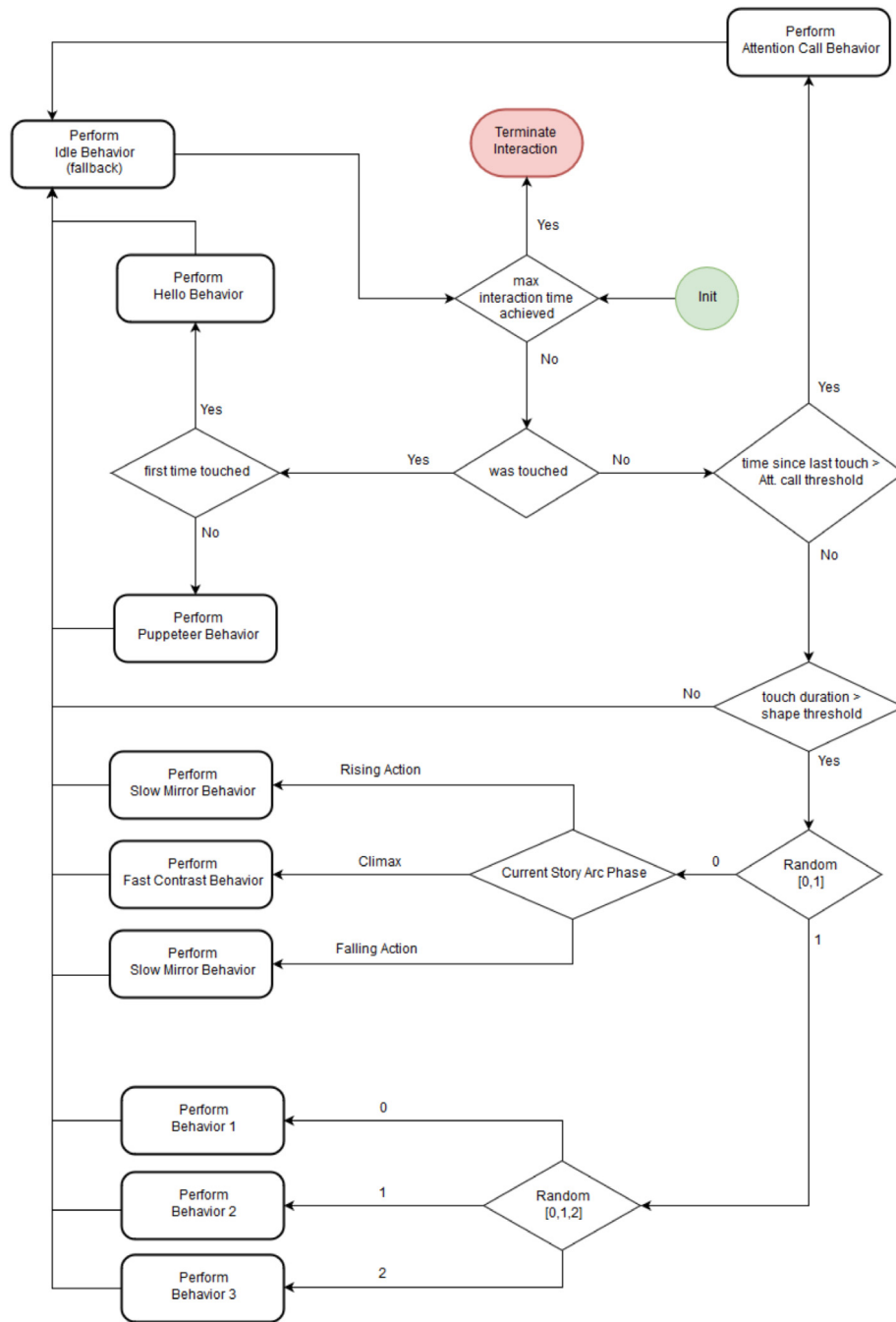

Fig. 5. State machine diagram representing the schedule of the procedures executed in the *Planning* module.

- **Falling action —** The story shifts to an action that happens because of the climax, which means that the conflict is resolved and the story reaches its end. During this stage, YOLO stimulates convergent thinking by using the mirror creativity technique (see Section 4.2).

4.4. Social behavior

YOLO expresses different social profiles to exhibit social behaviors. The profiles are named *Exuberant*, *Aloof*, and *Harmonious*.[9] These social behaviors appear as pre-sets when YOLO is

turned on and can be used interchangeably, making the robot a flexible character in the children's stories. The three different social modes for YOLO are explained below:

- **Exuberant —** YOLO reacts to every social interaction in an "enthusiastic" manner. Movements are fast and have a high amplitude. It displays vibrant purple and red colors with high brightness levels. As Exuberant, YOLO is proactive and seeks out social interaction. This is a vibrant, frenetic, and daring social profile;
- **Aloof —** YOLO is less "socially reactive" and is a "shy robot". In this mode, the robot exhibits low amplitude, slow movements and displays cold colors such as green and blue with low brightness levels. As Aloof, YOLO is not proactive; does

---

[9] Parameterization specifications for all the social behaviors of YOLO can be found here: https://github.com/patricialvesoliveira/YOLO-Software/wiki/SocialProfile.
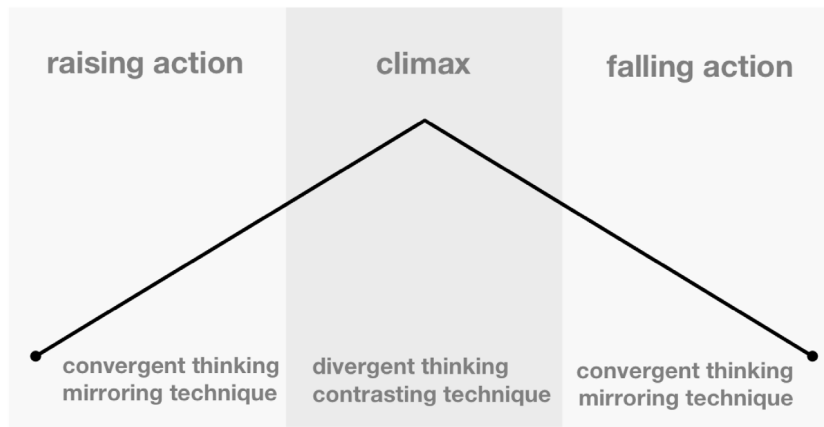
Fig. 6. Storytelling arcs associated with creativity techniques: convergent thinking is stimulated during rising and falling action phases by using the mirror technique; divergent thinking is stimulated during climax by using the contrast technique.

not seek interactions. This profile could also be described as loner, contemplative, or reclusive;

- **Harmonious —** YOLO acts in a moderated fashion, presenting behaviors that are in-between the extreme versions of Exuberant and Aloof. As Harmonious, YOLO exhibits medium speed, movements with medium amplitude, and displays warm colors such as yellow and orange at medium brightness levels. This is a balanced and moderate profile.

4.5. Child–robot interaction design

Children use YOLO as a character in their stories, moving it around while creating story-lines. The robot, being an active character, provides ideas for new narratives. It does so by using two techniques aimed at stimulating divergent and convergent thinking, "Mirroring" and "Contrasting", respectively. The robot decides which technique to use according to the story arc in which children are involved (see Fig. 6). Each story arc is associated with fixed timing, e.g., Raising Action starts at time 0 and goes until time 5. The timings were defined given knowledge collected from previous studies where children created stories with the same robot with the purpose of establishing a mean time for each story arc. Additionally, YOLO can be puppeteered by children mimicking a traditional toy. The puppeteer mode is activated when YOLO detects children's physical contact. Therefore, while children are physically touching the robot, i.e., playing, the robot refrains from performing any movement. This mode enables children to be in full control of YOLO, thus, maintaining traditional play formats.

5. Illustrative example

To validate the effectiveness of our software, we have tested it with children in a storytelling activity. The instruction provided information that they should use the robot as a character for the story they created. In the box below, we transcribed part of an interaction case during a study session between a child and YOLO (see complementary Fig. 7). In this example, it is visible how the robot makes use of its interaction profiles to stimulate convergent and divergent thinking and how this relates to the different stages of the storytelling.

The child is on the floor playing with YOLO.
Child: *"This is a football field and YOLO is from the Juventus team, so we are going to win!"*
The child manipulates YOLO in the imaginary football field, imitating the robot running after an imaginary ball and deviating from imaginary team adversaries. Because YOLO is still in the first part of the storytelling arc, i.e., in the Raising Action stage, the robot will stimulate convergent thinking abilities. Therefore, the robot imitates the last movement that the child performed. The child looks at the robot while it is moving.
Child: *"Yes! Go for it, Ronaldo, score!* (Ronaldo is the name of a Portuguese football player).
The child imitates scoring a goal and then grabs YOLO and celebrates.
Child: *"Ok Ronaldo, but we have to continue doing well. These other guys are good too.*
The child continues manipulating YOLO through the adversaries. At this point in time, YOLO entered the next storytelling arc which is the Climax. During climax, divergent thinking is stimulated so the robot will perform a movement that is different from the last movement that the child has performed. The child manipulates the robot straight ahead towards the soccer goal but the robot goes the opposite direction.
Child: *"What happened? Oh no, the other guys hit you in the knee. Assistance is needed here!"*
The game continues.

---

6. Conclusions and impact

As societies develop increasingly higher levels of sophistication, social robots can play a crucial role in the development of human creativity. Related research has indicated that social robots impact the play behaviors in children, pulling them towards traditional play formats such as physical, unstructured, and unrestrained play, benefiting multiple aspects of growth [34].

In this article, we presented the software which allows the YOLO robot to encourage creativity stimulation. This software allows potential developers to create behaviors that make the robot act according to different social behaviors. We also described several tests applying our software in real-world scenarios. These tests revealed the potential of our program, as robots using our software provoke creative narratives in stories which the children created.

Fig. 7. Use-case example of a child using YOLO as a character for the creativity-stimulating storytelling scenario.

The impact of this software is broad. By being an easy-to-use tool, stakeholders such as educators and parents have access to a robot that is easy to prepare (e.g., for Science, Technology, Engineering, Art, and Mathematics (STEAM)-related activities), contrasting with other existing technological tools that can be cumbersome for non-experts to prepare [19]. Additionally, this software serves as a solid platform in academic studies, where researchers can use YOLO's API to study child–robot interaction.

7. Highlights

- **Creativity stimulating robot:** YOLO can stimulate children's creativity during play;
- **Open-access:** Access to the code and the guide to install and execute the software;
- **Scalability and personalization:** API for developers to create new behaviors for YOLO;
- **Application:** YOLO can be used by children's stakeholders and by the research community.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

References

[1] Robinson K. Out of our minds: Learning to be creative. Mankato, MN, USA: Capstone; 2011.
[2] Collard P, Looney J. Nurturing creativity in education. Eur J Educ 2014;49(3):348–64. http://dx.doi.org/10.1111/ejed.12090.
[3] Baer J. Content matters: Why nurturing creativity is so different in different domains. In: Beghetto RA, Sriraman B, editors. Creative contradictions in education. Switzerland, Cham: Springer; 2017, p. 129–40.
[4] Kaufman J. Creativity as a stepping stone toward a brighter future. J Intell 2018;6(2):2–7. http://dx.doi.org/10.3390/jintelligence6020021.
[5] Gardner H, Gardner E. Art, mind, and brain: A cognitive approach to creativity. New York, NY, USA: Basic Books; 2008.
[6] Scott G, Leritz LE, Mumford MD. Types of creativity training: Approaches and their effectiveness. J Creat Behav 2004;38(3):149–79. http://dx.doi.org/10.1002/j.2162-6057.2004.tb01238.x.
[7] Ma H-H. The effect size of variables associated with creativity: A meta-analysis. Creat Res J 2009;21(1):30–42. http://dx.doi.org/10.1080/10400410802633400.
[8] Shneiderman B. Creativity support tools: A grand challenge for HCI researchers. In: M. R, C. B, M. O, editors. Engineering the user interface. London, UK: Springer; 2009, p. 1–9. http://dx.doi.org/10.1007/978-1-84800-136-7_1.
[9] Pires A, Alves-Oliveira P, Arriaga P, Martinho C. Cubus: Autonomous embodied characters to stimulate creative idea generation in groups of children. In: International conference on intelligent virtual agents. Lecture notes in computer science, vol. 10498, Cham: Springer; 2017, p. 360–73. http://dx.doi.org/10.1007/978-3-319-67401-8_46.
[10] Smith GF. Idea-generation techniques: A formulary of active ingredients. J Creat Behav 1998;32(2):107–34. http://dx.doi.org/10.1002/j.2162-6057.1998.tb00810.x.
[11] John OP, Srivastava S, et al. The big five trait taxonomy: History, measurement, and theoretical perspectives. In: Handbook of personality: Theory and research, Vol. 2. 2nd ed.. New York and London, US and UK: The Guilford Press; 1999, p. 102–38.
[12] Alves-Oliveira P, Arriaga P, Paiva A, Hoffman G. Yolo, a robot for creativity: A co-design study with children. In: Proceedings of the 2017 conference on interaction design and children. ACM; 2017, p. 423–9. http://dx.doi.org/10.1145/3078072.3084304.
[13] Alves-Oliveira P, Arriaga P, Paiva A, Hoffman G. Guide to build yolo, a creativity-stimulating robot for children. HardwareX 2019;6. e00074. http://dx.doi.org/10.1016/j.ohx.2019.e00074.
[14] Florida R. Cities and the creative class. New York: Routledge; 2005.
[15] Kim KH. The creativity crisis: The decrease in creative thinking scores on the torrance tests of creative thinking. Creat Res J 2011;23(4):285–95. http://dx.doi.org/10.1080/10400419.2011.627805.
[16] Runco MA. Everyone has creative potential. In: Creativity: From potential to realization. Washington, DC, US: American Psychological Association; 2004, p. 21–30. http://dx.doi.org/10.1037/10692-002.
[17] Sawyer RK, Csikszentmihalyi M, John-Steiner V, Moran S, Feldman DH, Gardner H, et al. Creativity and development. Counterpoints: Cognition, Memo; 2003.
[18] Di Blas N, Paolini P, Sabiescu A. Collective digital storytelling at school as a whole-class interaction. In: Proceedings of the 9th international conference on interaction design and children. ACM; 2010, p. 11–9. http://dx.doi.org/10.1145/1810543.1810546.
[19] Chan S, Yuen M. Personal and environmental factors affecting teachers' creativity-fostering practices in Hong Kong. Think Skills Creat 2014;12:69–77. http://dx.doi.org/10.1016/j.tsc.2014.02.003.
[20] Breazeal CL. Designing sociable robots. Cambridge, MA, US: MIT Press; 2004.
[21] Reeves B, Nass C. How people treat computers, television, and new media like real people and places. Stanford and Cambridge, CA and MA, US: CSLI Publications and Cambridge University Press; 1996.
[22] Ali S, Moroso T, Breazeal C. Can children learn creativity from a social robot? In: Proceedings of the 2019 on creativity and cognition. ACM; 2019, p. 359–68.
[23] Gordon G, Breazeal C, Engel S. Can children catch curiosity from a social robot? In: 2015 10th ACM/IEEE international conference on human-robot interaction (HRI). IEEE; 2015, p. 91–8. http://dx.doi.org/10.1145/2696454.2696469.

[24] Altman NS. An introduction to kernel and nearest-neighbor nonparametric regression. Amer Statist 1992;46(3):175–85.

[25] Barber CB, Dobkin DP, Dobkin DP, Huhdanpaa H. The quickhull algorithm for convex hulls. ACM Trans Math Softw 1996;22(4):469–83. http://dx.doi.org/10.1145/235815.235821.

[26] Alrutz M. Digital storytelling and youth: Toward critically engaged praxis. Youth Theatr J 2015;29(1):1–14. http://dx.doi.org/10.1080/08929092.2015.1020184.

[27] Brenner W, Uebernickel F, Abrell T. Design thinking as mindset, process, and toolbox. In: Design thinking for innovation. Cham: Springer; 2016, p. 3–21. http://dx.doi.org/10.1007/978-3-319-26100-3_1.

[28] Elbow P. Teaching thinking by teaching writing. Change: Mag High Learn 1983;15(6):37–40. http://dx.doi.org/10.1080/00091383.1983.10570005.

[29] Rickards T. Problem-solving through creative analysis. Wiley; 1975, http://dx.doi.org/10.1057/jors.1975.185.

[30] VanGundy AB. Techniques of structured problem solving. Springer; 1988.

[31] Freytag G. Die technik des dramas. Hirzel; 1872.

[32] Freytag G. Freytag's technique of the drama: an exposition of dramatic composition and art. Scholarly Press; 1896.

[33] Wright A. Storytelling with children. Oxford University Press; 1995.

[34] Pellegrini AD, Smith PK. Physical activity play: The nature and function of a neglected aspect of play. Child Dev 1998;69(3):577–98. http://dx.doi.org/10.1111/j.1467-8624.1998.tb06226.x.