# Learning Resources with Augmented Reality

**Lázaro V. O. Lima**
Centro ALGORITMI, Universidade do Minho, Braga, Portugal
lazaro.lima@ifb.edu.br

**Cristiana Araújo**
Centro ALGORITMI, Universidade do Minho, Braga, Portugal
decristianaaraujo@hotmail.com

**Luis Gonzaga Magalhães**
Centro ALGORITMI, Universidade do Minho, Braga, Portugal
lmagalhaes@dsi.uminho.pt

**Pedro R. Henriques**
Centro ALGORITMI, Universidade do Minho, Braga, Portugal
prh@di.uminho.pt

**Abstract**

Preparing teachers and students for a connected and programmed world depends on how we develop and reinvent teaching tools. The society has realized and is absorbing Computational Thinking and its related skills. The pragmatics shows that a person only acquires a new way of thinking or a new way of behaving if he is trained with the appropriate devices. Computational Thinking should be training from an early age to acquire important skills; in that way, the interpretation and design of algorithms/programs will become much easier. However, the development of Computational Thinking requires the creation and use of appropriate Learning Resources (LR). We will discuss how an ontology can be used to specify what is involved in Computer Programming and how these concepts and Computational Thinking concepts are related. We believe that this formal description will guide the choice of convenient LR. In that context, we intend to investigate the impact of Augmented Reality on them. After presenting the ontological approach, the paper will focus on the process of shaping Computational Thinking through Augmented Reality. We aim at creating AR-based LR prototypes to validate the idea we present here. We are convinced that an attractive way to improve fundamental skills is necessary to practice and use these tools with young students, but LRs must be attractive, motivating and effective.

**2012 ACM Subject Classification** Computing methodologies → Mixed / augmented reality

**Keywords and phrases** Computational Thinking, Learning Resource, Augmented Reality, Teacher Support Tools

**Digital Object Identifier** 10.4230/OASIcs.ICPEC.2020.15

**Funding** This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020.

## 1 Introduction

To promote the development of fundamental skills in the students like reading, writing, arithmetic, or analytic capabilities, nowadays we need to adopt new strategies to teaching and learning process.

These skills are crucial for many activities that in general require Problem-Solving capabilities as it is the case of Computer-based tasks demanding for Computational Thinking (CT) ability. In that direction, CT shall be included as a fundamental skill in the school curricula. The aim of such a decision is to develop in the student competencies for problem-solving that will be required to the 21st-century citizens. To train and induce CT in the student a novel teaching/learning process must be devised using techniques derived from

Mathematics, Gaming, and Computer Science. The abilities that characterize CT – like logic reasoning, abstraction, rigor in analysis and specification, strategic planning, etc. – are of uttermost relevance in Programming. Augmented Reality(AR) can provides greater motivation, gains learning, and delights the students who use it. AR is defined by Azuma [3]as the overlapping of virtual information in the real world through technology. This information can be simple textual images or 3D objects. Unlike Virtual Reality where the user is fully immersed in the environment and visual sense is controlled by the system, AR increases information in the real world, the user maintains a sense of presence in the real world and requires mechanisms to combine the real world with the virtual one. Unlike Virtual Reality consists in the simulation of virtual scenes, raising the user to an experience of immersion and interaction in a virtual world based on simulation generated by computer. AR supports pedagogical approaches through constructivism learning by enabling educational experiments that complement the activities of the real classroom like [24], one of the works that explore AR as a pedagogical tool. We believe that it is possible to explore AR as a technology that provides constructs to develop skills of uttermost importance for computer programming, not only as a mere technology operator, but also as a computationally literate individual. AR in education can be applied in the training of students' abilities, encouraging learning based on discoveries. Summing up, AR can be used to provide a rich contextual learning environment, adhering to constructivist principles, fostering opportunities for multiple learning styles, engaging learners in ways that are not possible in real-world without real consequences if mistakes are made during the training. These advantages will be used to created LR that should be available to promote CT in schools. Our goal is to show that we can mix technological facilities for the creation of new tools such as Augmented Reality.

This paper is organized in five sections, Section 2, with objectives and Research Methodology, Section 3 Computational Thinking is described, in Section 4, *OntoCnE*, a Ontology to describes the Computational Thinking domain, in Section 5, the work in progress and Section 6 for conclusions paper.

## 2 Objectives and Research Methodology

The information in the AR is apprised in real-time that provides an increase in the attention of the students [15]. Thus the AR can be a powerful allied technology in the development of CT in students exploring in different ways the decomposition, pattern recognition, abstraction and algorithm construction[7]. With the popularization of games and applications that use AR, there is also the adaptation for different platforms, including mobile platforms, in which the use of mobile phones to aid in education using AR is possible today.

We will use OntoCnE ontology to describe CT. This ontology is discussed with more details in the works of Araújo [1]. Consequently, the main objective of the research is improve motivation in teaching CT, creating atool that with formal descriptions of an Ontology-driven Learning Resource designed to describe CT will result in LR using Augmented Reality techniques.

The methodology used in this research to achieve our objectives will be Design Science Research, this methodology focuses on the development and performance of artifacts with explicit intentions of functional improvement of the developed artifact. DSR is most commonly applied in the development of artifacts such as algorithms, interfaces, methodologies design, languages. DSR's focus is to develop the knowledge to design solutions to problems in a particular field according to [20]. The DSR consists of a sequence of activities that produces an innovative product, with the artifact created it is possible for the researcher to better understand the problem and have a better view to reassess the problem and thus improve the quality of the design process in a construction loop until developing a final artifact to solve complex and relevant field problems. Because the focus of this methodology is on creating an artifact for a given problem, this artifact must be well evaluated to ensure its objectives. It also should solve a problem that has not been resolved or provide a better solution. For developing this artifact we use the DSR methodology, Hevner [10] counts with seven guidelines to follow: Problem identification and Motivation; Objectives of a Solution (research to define objectives); Design and Development (creation of the artifact); Demonstration (used in appropriate environment); Evaluation (performance of the artifact); Communication.

## 3 The Importance of Teaching Computational Thinking

In 2006, Wing [21, 22, 23] proposed the foundations of CT and showed how society is influenced by technology even more in education. According to the author, the most important and high level thinking process is the process of abstraction, being used in the definition of patterns, generalizing from specific instances and parametrization. CT is a method for solving problems, or designing systems and understanding human behavior, based on the fundamental concepts of computer science, that develop competencies in students required in the 21st century.

Computational Thinking is based on the concepts of Pattern Recognition, Abstraction, Problem Decomposition, Algorithms. Moreover students acquiring those skills to solve problems, are also able to debug and assess the calibration of the proposed solution. In terms of the use of technological resources, in order to introduce the concepts of programming languages, it is important to pay attention that they must be able to motivate and encourage the students, development of abstraction, decomposition of problems and the organization of steps to solve a problem. Moreover they should allow for a constructivist-based teaching approach, that has been proved to promote effective knowledge acquisition. The ability to formulate algorithms for computers is like building instructions for a computer to solve / repeat processes; this action is related to solving simple or complex tasks, but learning how to build algorithms has a great cognitive load and needs to be trained since young as explained in the work done in an effort to incorporate CT in curricula[16].

With adequate resources it is possible to work with the identification of common characteristics between the problems and their solutions. We can further identify patterns among the sub-problems that have been abstracted, finding an efficient solution to the problems encountered. It is also possible to work with resources that help breakdown processes in smaller parts for easier resolution. A learning activity can use a LR as an unplugged activity or a game as demonstrated in work of [11]. Thus it is possible to prepare the thought so that it arrives at the moment of creation of the Algorithms in the strategy or clear instructions for the solution of the problem.

Resnick [18] explores CT, but the use of AR can be observed in the works of [17, 12]. The exploration of [12] takes the earlier work of *CodyRoby* into a low-cost AR mobile system, which uses a simple smartphone as an augmented sensor to transform a fully disconnected coding set into an Augmented Reality coding experiment. The relationship between the development of CT and programming learning with AR can also be seen currently in the investigation of [13, 19, 8].

## 4 Describing Computational Thinking with an Ontology

An Ontology, in Computer Science, represents a set of concepts within a domain and the relationships between them. *An ontology is an explicit specification of a conceptualization* [9]. Furthermore used to represent knowledge and to perform inference on domain objects. We felt the need for a formal definition of the domain we are coping with. In that sense we decided to describe it creating a specific ontology that describes the domain of CT in [1].

The ontology we create, which is called *OntoCnE* [1], describes the CT domain, more specifically how to teach it and what material is needed to teach it in the various years of schooling. After the construction of the ontology, we select the concepts that would be taught in each school year and add new concepts to ontology. This ontology will allow to classify the resources that will be used to train a certain concept at a given level of education. The research proposed by Azevedo [2] describes Micas, as a tool that allows to store the resources and classifies them according to the *OntoCnE*, the tool can be accessed at `https://micas.epl.di.uminho.pt/`. After getting to know a part of *OntoCnE*, in the next section we will present how the working tool will be developed. In the following fragment, we can see concepts taught in the 1st year of *OntoCnE*.

**Listing 1** Concepts taught in the 1st year (fragment).

```
Triplos{
ano1 =[
    desenvolve=> PensamentoComputacional,
    desenvolve=> RaciocinioLogico,
    desenvolve=> Abstraccao,
    apresenta=> Problema,
    introduz=> Algoritmo,
    introduz=> Instrucao,
    introduz=> Programa,
    introduz=> DispositivoDigital,
    introduz=> LingGrafica,
    usa=> Computador,
    usa=> Robot ];
}
```

## 5 Learning Resources with Augmented Reality

Learning Resources is a tool that helps teachers in teaching and student on learning. LR are hard or soft devices that allow students to train previous knowledge or acquire new knowledge, stimulating their ability to comprehend, organize and synthesize educational content in a specific domain [4]. The LR can be simple and developed based on drawn letters or printed at home, demonstrating simplicity and accessibility for use. There are activities directly linked to programming logic as activities related to loops, sequences, events, conditionals, working with binary numbers, or even activities directly linked to the training of CT.

[1] From the Portuguese *Ontology for Computation in School*

There are *Code.org* [2], activities like *CT with Monsters* in [14] can be adapted, task focused on decomposition, then students will analyze a catalog of monsters for patterns, abstract similar details from the monsters, then use that information to create an algorithm (instructions) for other students to draw a certain monster. Students can alternate algorithms with another group and test to see each others result (Debug).

It is crucial to have adequate resources to train the different skills involved in CT. The more resourceful, motivating and effective, the better students will shape their minds by learning the skills they desire.

Our goal with this paper is to show that we can mix a technological facility, AR, to build Learning tools that as the literature describes, LR with AR will increase student motivation as [6, 5] shows in his work, applying AR resource as a teaching tool not only can create a learning environment. Bearing in mind that the generated AR-LR should work in the Web Browser or on other platforms that do not require high computing power neither and complex, expensive operating equipment to be purchased by schools.

With a work environment formed with a generated AR-LR, having as one of its goals, represent analogies to understand complicated programming concepts. This definition, ruleset, and operation step using *OntoCnE* will allow us to quickly generate and modify the new AR-LR.

In the last step, we will study how to start generating a description to later generate new artifacts, with functionalities integrated with real activities and alternative technologies. These artifacts will provide the path to create the appropriate LR according to the description of Ontology for CT. Thus it is possible for students to understand the general levels of programming and how to think algorithmically to arrive at a solution. Hence understanding the concepts necessary for CT. The construction of such a tool will also focus on usability according to the studies reported in previous section. The tasks performed by students in LR activities can not be difficult or too easy to avoid disinterest in students.

A first proposal of the system architecture is depicted in Figure 1, where can be seen that the main users of the system will be the teacher and the resources development. The teacher using an appropriate tool will have an impruve in his learning activities. The methodology adopted prescribe the execution of a sequence of activities that produces an innovative product. With the created artifact, it is possible for the User and the Researcher to better understand the problem and have a better view to reassess the problem and, thus, improve the quality of the design process in construction.

The artifacts to be developed aiming to impact on the development of Computational Thinking using Augmented Reality, shall assure that the interaction with the virtual objects will be during the use of the tool, not just showing a simple 3D object to the student. The participation of the teachers or future users of the system in the process of construction of the functional requirements is primordial. The description of CT through the ontological approach, will be carried out by *OntoCnE*. Libraries that allow the use of AR, integrate the system in order to be included into the 3D Object behavior rules.

Then exemplifying the ideas, in figure 2 we can observe the possible interactions, the student changes the properties of 3D objects and visualizes 3D geometric shapes in AR. Thus, we have the introduction of concepts of CT and programming using Spatial Geometry with AR, in which students have difficulty in having the abstract notion of geometric figures.

First using *OntoCnE* to describe the CT. The architecture will be defined to develop AR artifacts. Micas be used to create an LR repository to make it available to teachers. *OntoCnE* describes the domain of CT and the objective is to assist in the classification

[2] `https://code.org/curriculum/unplugged`

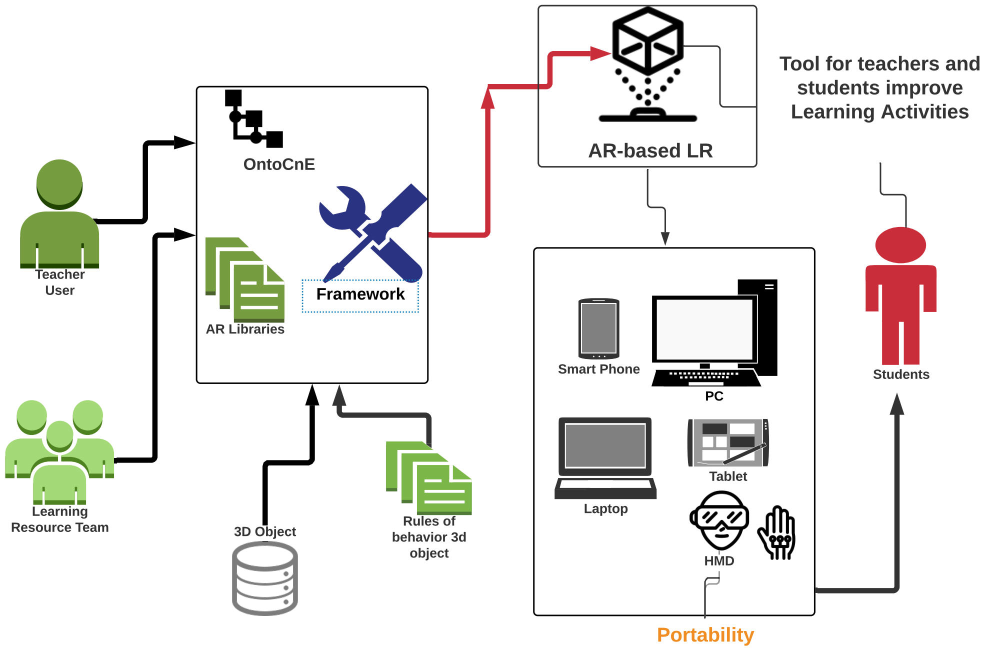


**Figure 1** Proposed Scheme to generate AR-LR.

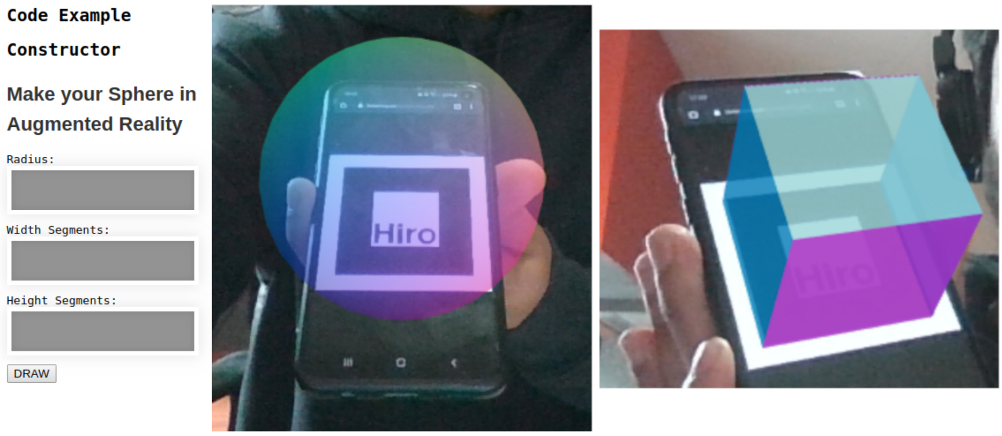


**Figure 2** Example of the interaction in the prototype under construction.

of LR. It is important to highlight that we want to demonstrate that Augmented Reality contributes to the development of skills related to Computational Thinking, so Plugged Learning Resources will be generated. We see that the main users of the system will be the teacher and the LR development. The tasks performed by students in LR activities can not be difficult or too easy to avoid disinterest in students. After performing the activities related to the objectives of the artifact, a final survey will be verified, the improvement in the skills or not through the analysis of the results.

## 6 Conclusion

To learn Computer Programming is necessary to analyze a problem, to design solutions, test and optimize solutions, not only coding. Computational Thinking helps expressing how to solve a problem. CT involves concepts like decomposition, abstraction and algorithmic design. With those ingredients, CT potentiates the ability to program computers. Learning Resources are crucial to train properly CT, so the more complete and wiser they are, the more effective their help. We believe that smart choices must be made to create adequate LRs. In that context, we research the inclusion of Augmented Reality components in some devices to create new, improved, LRs.Producing those Augmented LR is challenging and an interesting task but time consuming if done manually. So in this paper we suggested the use of a generation mechanism capable of producing the required resources. The process will be guided by *OntoCnE*, an ontology for CT, to provide a formal representation of the knowledge domain. That automation platform will leverage the production and availability of the adequate effective resources. However, in order to evaluate the impact of Augmented LRs on CT, we will design and conduct experiments with real students in real classrooms to measure the results in learning activities with and without AR. AR technology can provide animation, sound, and video to make traditional resources more appealing, and more helpful transmitting information.. Guided by OntoCnE, the prototype will have the combination of introduction to programming, presenting definitions and properties, and teaching spatial geometry in mathematics. Using Augmented Reality to interact with the created 3D objects.

The ongoing work is devoted to build prototypes to validate the idea presented here. To get the most out of CT skills, we are convinced that it is necessary to practice and use well chosen Learning Resources, but LRs must be attractive, motivating and effective.

## References


1 Cristiana Araújo, Lázaro Lima, and Pedro Rangel Henriques. An Ontology based approach to teach Computational Thinking. In Célio Gonçalo Marques, Isabel Pereira, and Diana Pérez, editors, *21st International Symposium on Computers in Education (SIIE)*, pages 1–6. IEEE Xplore, November 2019. doi:10.1109/SIIE48397.2019.8970131.

2 Ana Azevedo, Cristiana Araújo, and Pedro Rangel Henriques. Micas, a Web Platform to Support Teachers of Computing at School. In A. J. Osório, M. J. Gomes, and A. L. Valente, editors, *Challenges 2019: Desafios da Inteligência Artificial, Artificial Intelligence Challenges*, pages 625–641. Universidade do Minho. Centro de Competência, 2019-05.

3 Ronald T. Azuma. A survey of augmented reality. *Presence: Teleoperators and Virtual Environments*, 6(4):355–385, 1997. doi:10.1162/pres.1997.6.4.355.

4 Rona Bušljeta. Effective use of teaching and learning resources. *Czech-Polish Historical and Pedagogical Journal*, 5(2):55–70, 2013.

5 Su Cai, Enrui Liu, Yang Yang, and Jyh-Chong Liang. Tablet-based ar technology: Impacts on students' conceptions and approaches to learning mathematics according to their self-efficacy. *British Journal of Educational Technology*, 50(1):248–263, 2019.

6 Wen-Hung Chao and Rong-Chi Chang. Using augmented reality to enhance and engage students in learning mathematics. *Advances in Social Sciences Research Journal*, 5(12), 2018.

7 Cheng-Yu Chung and I-Han Hsiao. An exploratory study of augmented embodiment for computational thinking. In *Proceedings of the 24th International Conference on Intelligent User Interfaces: Companion*, pages 37–38. ACM, 2019.

8 Xiaozhou Deng, Qiao Jin, Danli Wang, and Fang Sun. Arcat: A tangible programming tool for dfs algorithm teaching. In *Proceedings of the 18th ACM International Conference on Interaction Design and Children*, pages 533–537. ACM, 2019.

9 Thomas R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. In *International Journal of Human-Computer Studies*, pages 907–928. Kluwer Academic Publishers, 1993.

10 Alan R Hevner, Salvatore T March, Jinsoo Park, and Sudha Ram. Design science in information systems research. *Management Information Systems Quarterly*, 28(1):6, 2008.

11 Cagin Kazimoglu, Mary Kiernan, Liz Bacon, and Lachlan Mackinnon. A serious game for developing computational thinking and learning introductory computer programming. *Procedia-Social and Behavioral Sciences*, 47:1991–1999, 2012.

12 L Klopfenstein, Andriy Fedosyeyev, and Alessandro Bogliolo. Bringing an unplugged coding card game to augmented reality. *INTED Proceedings*, pages 9800–9805, 2017.

13 Divna Krpan, Saša Mladenović, and Biserka Ujević. Tangible programming with augmented reality. In *12th International Technology, Education and Development Conference*, 2018.

14 CODE ORG. Available in `https://code.org/`, 2015. Access date: set 2019.

15 Mark Petrorvich, Mamta Shah, and Aroutis Foster. Augmented Reality Experiences in Informal Education. In *2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, pages 815–819, 2019. `doi:10.1109/tale.2018.8615396`.

16 Jake A Qualls and Linda B Sherrell. Why computational thinking should be integrated into the curriculum. *Journal of Computing Sciences in Colleges*, 25(5):66–71, 2010.

17 Iulian Radu and Blair MacIntyre. Augmented-reality scratch: a children's authoring environment for augmented-reality experiences. In *Proceedings of the 8th International Conference on Interaction Design and Children*, pages 210–213. ACM, 2009.

18 Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay S Silver, Brian Silverman, et al. Scratch: Programming for all. *Commun. Acm*, 52(11):60–67, 2009.

19 Chin-Hung Teng, Jr-Yi Chen, and Zhi-Hong Chen. Impact of augmented reality on programming language learning: Efficiency and perception. *Journal of Educational Computing Research*, 56(2):254–271, 2018.

20 Joan Ernst Van Aken. Management research as a design science: Articulating the research products of mode 2 knowledge production in management. *British journal of management*, 16(1):19–36, 2005.

21 Jeannette M Wing. Computational thinking. *Communications of the ACM*, 49(3):33–35, 2006.

22 Jeannette M. Wing. Computational thinking - What and why? *The Link Magazine*, 2011. URL: `http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why`.

23 Jeannette M. Wing. Computational thinking benefits society. *40th Anniversary Blog of Social Issues in Computing*, 2014(3):33, 2014. `doi:10.1145/1118178.1118215`.

24 Danny Yaroslavski. How does lightbot teach programming. *Retrieved January*, 29:2016, 2014.