2021 ASEE ANNUAL CONFERENCE
Virtual Meeting | July 26–29, 2021 | Pacific Daylight Time
Paper ID #32870

Work in Progress Pilot Study: Virtual Reality for Computational Thinking Foundations and STEM Enrichment

Dr. Katherine Levenick Shirey, EduKatey

Dr. Katey Shirey graduated from the University of Virginia with a B.A. in Physics and a B.A. in Sculpture (minor in art history). After teaching sculpture at UVA as an Aunspaugh Fellow, she completed her Masters of Teaching in secondary science also at UVA. Dr. Shirey taught high school physics in Arlington, VA, for five years and became a Knowles Teacher Initiative Teaching Fellow. During this time, she served as a teacher liaison to the IceCube Neutrino Telescope at the South Pole and was a NASA astronaut candidate finalist in 2013. Dr. Shirey earned her Ph.D. from the University of Maryland in 2017 after transitioning to study engineering integration in high school instruction as a site of creative thinking in physics learning. As founder and consultant for eduKatey, LLC, Dr. Shirey works with educators around the world to integrated science, technology, engineering, art, and mathematics content areas through curriculum development, professional learning, and research.

Dr. Magesh Chandramouli, Purdue University Northwest

Dr. Chandramouli is an Associate Professor of Computer Graphics Technology in Purdue University Northwest. Dr. Chandramouli has been invited to deliver keynote speeches and guest lectures in various countries around the world. Formerly a Frederick Andrews Fellow at Purdue University, West Lafayette, he completed his doctoral studies from the Department of Computer Graphics Technology. He completed Master of Engineering at the National University of Singapore and Master of Science from the University of Calgary, Canada. He completed his Bachelor of Engineering from the College of Engineering, Guindy, India. Dr. Chandramouli has published journal articles in prestigious international journals and has presented papers in respected national and international conferences. He has received federal, regional, and international grants for his work in areas including virtual reality, STEM education, Human Computer Interaction, and Genetic Algorithms in Graphics.

Pilot Study: Virtual Reality for Computational Thinking Foundations and STEM Enrichment (WIP)

Abstract
This paper presents the pilot study of a web-based desktop virtual reality (VR) instructional framework used to teach computational thinking (CT) concepts to secondary students. Classroom CT instructional practices are vastly underexplored in research on adolescent beginning programmers. Training in computational thinking, requires a firm grasp of various components ranging from fundamental aspects. The study's objective was to create a VR platform consisting of four VR learning modules to teach data types, conditionals, loops, and operators. Each module developed one CT topic with engaging interactive activities, animated models, and games with built-in self-assessment.

This paper details the modules' development, deployment, and outcomes related to the use of the VR modules within a science and math enrichment camp focused on learning engineering design and coding. The study assessed student use of the four CT topics in their final design project—a coded personal reflection. A lack of the fundamental understanding of CT concepts is a critical factor in STEM attrition rates as CT skills are highly interconnected to various branches of engineering and technology. So, we employ a CT perspective to deliver essential skills related to STEM concepts to facilitate skills transfer including problem solving and critical thinking. Students' final projects were analyzed including a block-coded animation or app in code.org and a written summary of the project, as well as an "artist statement" that was required to relate the CT topics to the project's program. Data analysis is still underway. Early conclusions indicate that explicit development of each CT topic was useful for project success if the coding platform also scaffolded coding using identical language as the modules (for loops to for loops, for example.) Potential impacts of this study include recommendations for introducing CT topics to high school beginning coders.

Introduction:
Several reform initiatives attempted to guide STEM education for American teachers and students. *A Framework for K-12 Science Education* [1] and the subsequent *Next Generation Science Standards* (NGSS) [2] include engineering and CT practices through the inclusion of "critical skills of mathematics," [3, p. 58]. The *Common Core Mathematics Standards* (CCSS) [4] compels connections to real-world problems that require "technological tools to explore and deepen their understanding of concepts" [4, p. 7]. These standards attempt to "ensure that students are equipped with the necessary knowledge and skills to be globally competitive" [5, p. 24]. To allow students to relate new learning to existing skills/knowledge without cognitive overload, teaching in technology environments should include as much contextual content as possible. CT is a natural ally to contextual technology and STEM education as it connects mathematics, science, computer science, and engineering content areas. Connections to mathematics and science in CT practices include appropriate variables, compositional reasoning, pattern matching, and procedural thinking [6]. CT concepts and CT skills are becoming ubiquitous in all branches of engineering and technology meaning that CT mastery is important for success in engineering education.

As of September 2020, forty-four states and the District of Columbia either use the NGSS standards now or use standards based on the NGSS, representing 71% of the nations' K-12 science students [7]. Forty-one states and the District of Columbia adhere to the CCSS [8]. Both NGSS and CCSS require learners to be capable of CT practices such as creating, using, and assessing data representation models. However, like many reform movements, it has been left to teachers, schools, and districts to implement CT-based instruction and survey data shows that the majority of K-12 teachers do not feel well prepared to teach computational thinking including breaking computer science problems into parts and using computational artifacts [9].

Virtual Reality Framework

Web-based VR environments show enormous promise for capturing teacher and student attention in PD and instruction. "VR is basically a way of simulating or replicating an environment three-dimensionally and giving the user a sense of being there, taking control, and personally interacting with that environment with his/her own body" [10]. We conducted an extensive literature review of studies involving VR in K-12 education spanning over the past two to three decades [11]–[16] and found positive effects of VR in education and training. The interactive visual learning methods designed in this study facilitate stimulating students' interest not only in CT related problems, but with overall STEM problem solving skills. The CT learning modules integrated three important constructs of VR: Visual approach, interactive game-based delivery, and differentiated instruction in a VR setting.

Visual approaches enable the construction of deep and new understandings [17]. Visual presentations can stimulate learners and aid information transfer to understand complex and abstract concepts better. Pocock [18] states that "Sight is, without doubt, our dominant sense, yielding nine-tenths of our knowledge of the external world." The positive effects of visual literacy [19] and visual skills in facilitating learning [20] support the need for visual aids with appropriate color-coded content [21]. VR has immense potential for providing visual learning opportunities, even in distance learning. One of the major advantages that virtual modules offer over conventional instructional practices is that they allow students to visualize the information presented in an interconnected, coherent manner rather than discrete or isolated chunks of data. This serves to effectually engage them and facilitates conveying CT concepts in an easily understandable manner.

Game-based approaches are wildly popular with youth and adults. 0% of the nation's video game players are in the age group of 18-35 years, 18% are between 36-49 years old, and 21% are under 18 years old [22]. Moreno and Mayer [23] defined "interactivity" as a two-way action (between learner and instructor) as opposed to a one-way action (i.e., from instructor to the learner). Boaler et al. [17] also corroborate the importance of manipulation and motion alongside visualization as an effective combination employed by teachers to communicate abstract concepts. Online learning modules should facilitate user interaction using intuitive HCI techniques to optimize learning, retaining, and applying CT concepts. While this works on some networked video game platforms, often the technology lift and privacy yield are too great to implement game-like interactivity in school settings.

When teaching fundamental concepts, 'cognitive overload' is a vital issue [24], [25]. Material complexity, failure to integrate new material with previous knowledge, and disruptive elements that distract from learning the materials can cause cognitive overload. In this research, a carefully designed set of visual exercises allows the user to interact and perform tasks that build on earlier learning fun and reduce cognitive overload. The UI design component [26] of the VR framework and HCI components will play a crucial role in understanding the materials presented.

Computational Thinking for Science and Mathematics

CT is a wrap-around term for the practices and conventions involved in creating useful, internally consistent data characterizations and data processing methods. This study employs interactive visual models to facilitate learning/training via practical and simple examples that students can relate to. Besides stimulating a diverse student body population's interest, such visual interactive learning allows them to explore and better understand the wide range of CT processes and concepts. CT is, "The thought processes involved in formulating a problem and expressing its solution in a way that a computer—human or machine—can effectively carry out" [6], [27]. As data and evidence "hold a primary position in deciding any issue" [28, p. 27] in science as well as in mathematics, the practices of CT are at the heart of how scientists and mathematicians deal with and use data.

Weintrop et al.'s [29] model for CT in mathematics and science includes four sets of practices: Data practices, modeling and simulation practices, computational problem-solving practices, and systems thinking practices. Table 1summarizes tasks in each area. We find CT in mathematical variables, compositional reasoning, pattern matching, and procedural thinking [6]. Data-informed models, conclusions, and predictions are foundational for science. Bridging through CT, mathematics, and science, teachers, could truly integrate STEM subjects in the content classroom. This study seeks opportunities to identify coding practices as evidence student learning in discussions, surveys, reflections, and coded artifacts.

Data Practices	Modeling and Simulation Practices	Computational Problem-Solving Practices	Systems Thinking Practices
• Collecting data • Creating data • Manipulating data • Analyzing data • Visualizing data	• Using computational models (CM) to understand a concept • Using CM to find and test solutions • Assessing CM • Designing CM • Constructing CM	• Preparing problems for computational solutions • Programming • Choosing effective computational tools • Assessing different approaches/ solutions to a problem • Developing modular computational solutions • Creating computational abstractions • Troubleshooting and debugging	• Investigating a complex system as a whole • Understanding the relationships within a system • Thinking in levels • Communicating information about a system • Defining systems and managing complexity

Table 1: *CT Practices for Science and Mathematics* [29]

Research Design:

This project delivered CT concepts during a STEM-enrichment summer camp using web-based VR. The platform was housed on a website that maintains interactivity and the required level of immersive experience without requiring specialized equipment or viewers. This study evaluated the CT VR as a learning tool for new coders by analyzing how the four CT topics presented in the VR were reflected in student's final projects when they had and had not interacted with the VR modules. Our research question was, how does student programming after interacting with CT VR compare with that of students who did not interact with the CT VR?

Of the 51 students in a 10-day online summer camp, 39 students and their parents consented to join this study. Of those 39, 35 completed the final project in the summer camp. Data was gathered from student interaction with the VR via pre- & post-VR interaction assessments, daily homework completion on code.org, a final coded project on code.org, and student notes to establish whether each student had or had not done the VR modules as homework. Google Classroom was used to disseminate and collect all assignments. The researchers accessed the student code on code.org through links shared by the students and teacher access. Student project outcomes include a coded artifact using block programming on code.org and written statements on a Google Slides presentation.

Implementation:

The students learned from two instructors in a completely online environment for 10 days including 30-50 minutes of synchronous instruction and one hour of asynchronous instruction daily. Using content-embedded curriculum and a blended learning approach, students completed two-day learning cycles including web-based VR CT learning modules and assignments on code.org. On the odd-numbered days, the students were introduced to the CT topics using VR modules, learned about the topics with the modules, and then practiced the application of each topic in code.org activities on the even-numbered day. Students showed their completion of the learning modules with screenshots and narrative summaries in daily typed notes files on Google Docs. Students showed their completion of code.org assignments by signing into code.org and completing assignments which the teacher could see through the code.org web interface.

The CT modules incorporated visual learning assets, including photorealism and a useful color-designed UI, to increase students' learning of STEM concepts. Some primary computational aspects we focus on in this study include data, variables, I/O (input/output), sorting, while the next intermediate steps include data abstraction, loops, iteration, algorithmic problem-solving, etc. Please see Figure 1 for sample screen shots from the CT VR designed for the camp. The VR strategies used in the VR learning modules capitalized on the gaming interests of adolescents and used a gaming inclination in its interactive VR modules.

The experience was safe. By maintaining sound user interface (UI) and human-computer interaction (HCI) principles, online students could actively participate in their learning and maintain safety and autonomy at a distance. Interactive pedagogical practices can reduce the cognitive load on students, creating a fun learning experience and reducing attrition. For instance, in this study, students are given contextualized and enriched examples for learning various programing concepts, which they can solve together with interaction and collaboration. Such activities can result in enhanced understanding and will facilitate experiential learning.

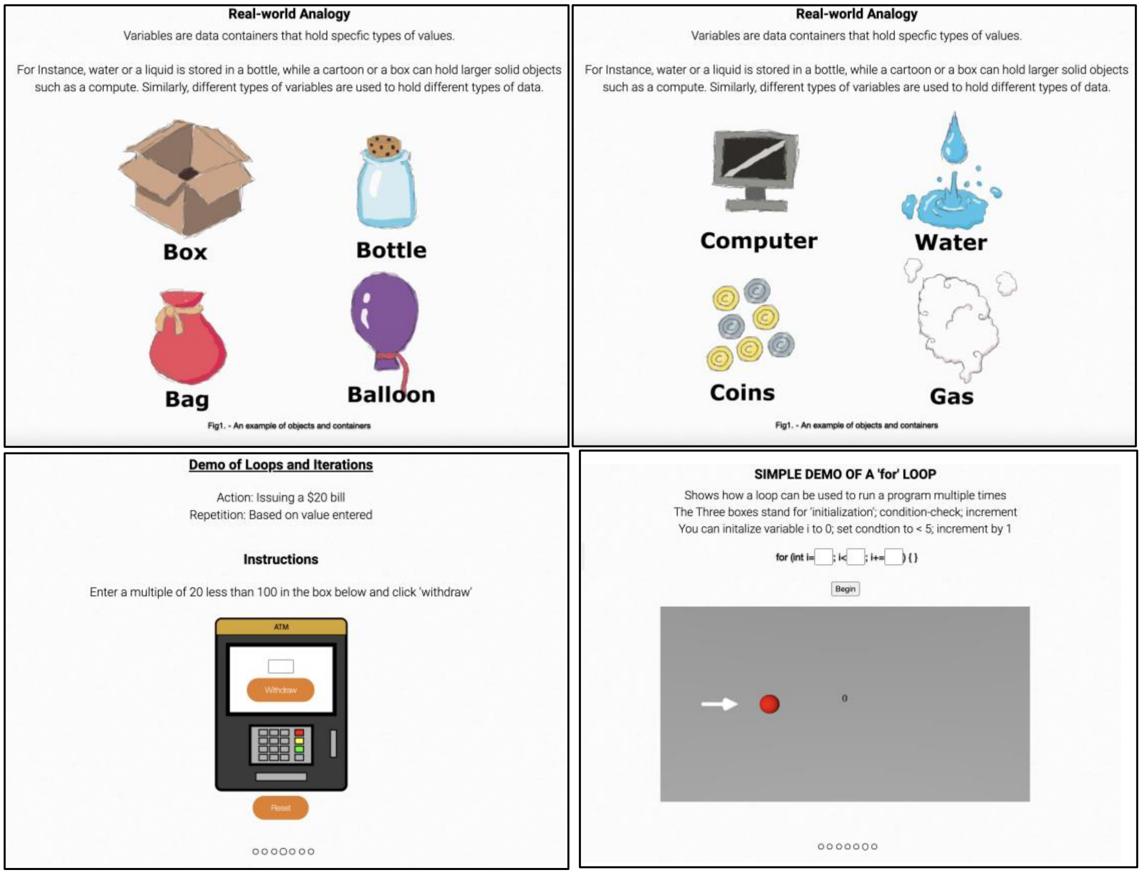

Figure 1. *CT VR Screenshots*

Summarily, the advantages of the interactive VR were:

- Ease of access, replicability, and dissemination to support nationwide STEM literacy
- Visualizing "what if?" scenarios impracticable in the real-world
- Cost reduction related to procurement & installation (compared to fully immersive VR)
- Enhanced safety of the participants/trainees due to the virtual nature of the training
- Minimal supervision due to reduced risk during virtual training

Results:

The design and implementation of the VR modules were done with the ultimate goals of serving the training, and dissemination process. The use of desktop-based virtual tools aided in the delivery of the learning materials facilitated accessibility and learnability. As this is a work-in-progress, comprehensive data analysis is currently in progress to understand how the student coding projects used the four VR CT topics and whether there was a difference between the coded products from students who used the VR versus those who did not use the VR (i.e., did not complete their homework.)

Potential implications of this study indicate that online web-based VR is readily accessible for multiple platforms of deployment. Educators should consider the ability of web-based VR with mobile adaptability to engage students in learning CT content.

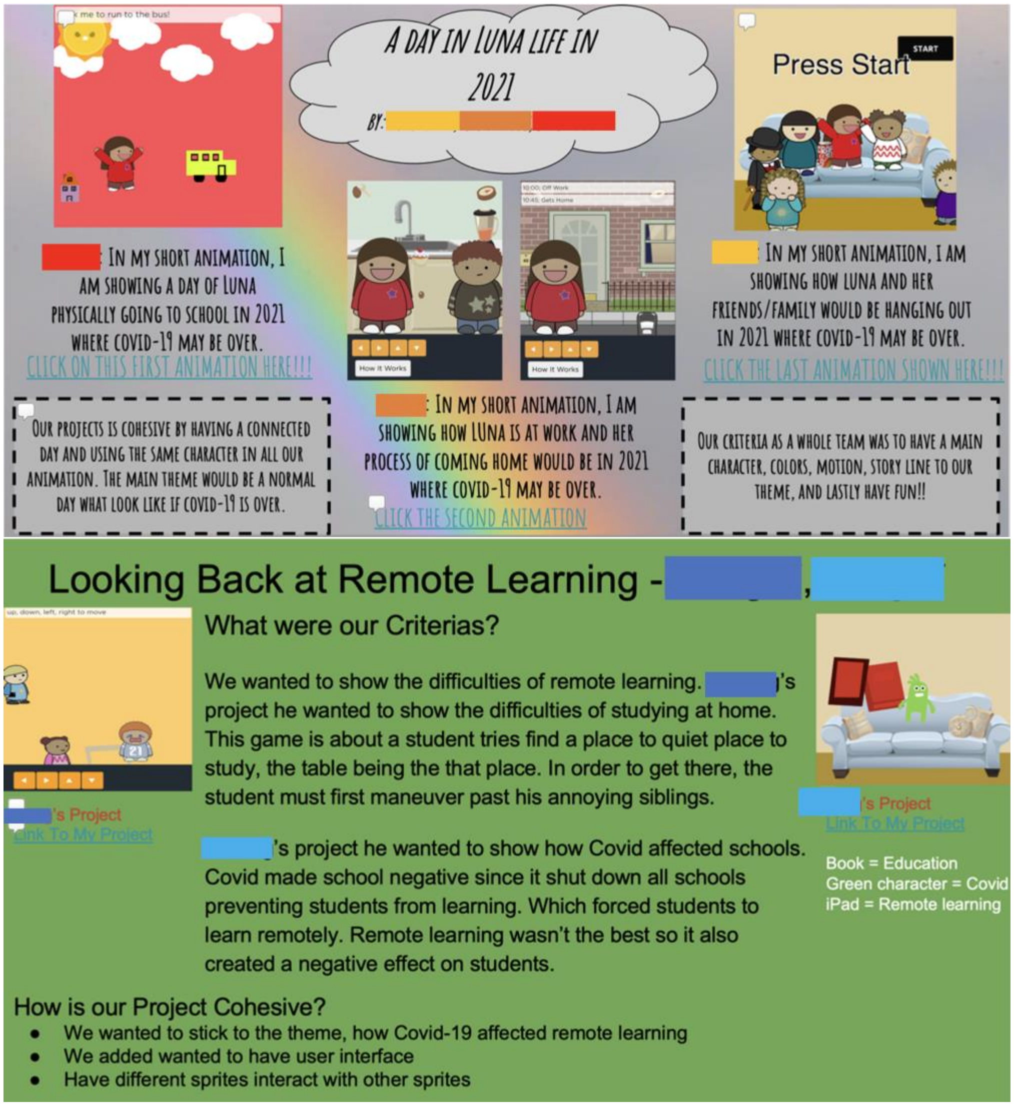

Figure 2. *Two sample showcase slides describing the individual coded projects and cohesive design elements.*

<u>Conclusion:</u>

Captivating students' attention necessitates connecting to the participant (student) in the learning process and VR is a proven tool that can engage learners effectively. The features of VR such as interaction and navigation facilitate actively engaging with the learning materials. The user can dynamically interact with the information for an engaging CT learning experience. While numerous opportunities are available for STEM students, higher-educational institutions are faced with the challenge of preparing students to take advantage of these opportunities. In this study, we employed the CT perspective to STEM teaching, as this integrates mathematics, science, computer science, and engineering content areas. The interactive visual learning methods designed in this study facilitate stimulating students' interest not only in CT related problems, but with overall STEM problem solving skills. One of the important reasons that contributes to increasing student drop-out rates and decreasing quality of STEM education is the over- reliance on conventional approaches that are ill-suited and outdated, which has in turn adversely affected students' attitude towards STEM learning.

References

[1] Committee on a Conceptual Framework for New K-12 Science Education Standards, *A framework for K-12 science education: Practices, crosscutting concepts, and core ideas*. The National Academies Press, 2012.

[2] NGSS Lead States, *Next generation science standards: For states, by states*. Washington, D.C.: The National Academies Press, 2013.

[3] NGSS Lead States, "Appendix F: Science and engineering practices," in *Next generation science standards: For states, by states*, no. April, Washington, D.C.: The National Academies Press, 2013, pp. 1–33.

[4] National Governors Association Center for Best Practices & Council of Chief State School Officers, *Common Core State Standards for Mathematics*. Washington, DC: National Governors Association Center for Best Practices and the Council of Chief State School Officers, 2010.

[5] C. D. Jerald, "Benchmarking for Success: Ensuring US Students Receive a World-Class Education.," *Natl. Governors Assoc.*, 2008.

[6] J. Wing, "Research notebook: Computational thinking—What and why," *link Mag.*, vol. 6, 2011.

[7] National Science Teaching Association, "About the Next Generation Science Standards." https://ngss.nsta.org/About.aspx (accessed Sep. 10, 2020).

[8] Common Core State Standards Initiative, "Standards in Your State." http://www.corestandards.org/standards-in-your-state/ (accessed Sep. 10, 2020).

[9] E. R. Banilower, P. S. Smith, K. A. Malzahn, C. L. Plumley, E. M. Gordon, and M. L. Hayes, "Report of the 2018 NSSME+," no. December, p. 442, 2018, [Online]. Available: http://www.horizon-research.com/report-of-the-2018-nssme.

[10] L. J. Ausburn and F. B. Ausburn, "Desktop virtual reality: A powerful new technology for teaching and research in industrial teacher education.," *J. Ind. Teach. Educ.*, vol. 41, no. 4, pp. 35–58, 2004.

[11] K. Fernie and J. D. Richards, *Creating and using virtual reality: a guide for the arts and humanities*. Oxford: Oxbow, 2003.

[12] B. Jordan and A. Henderson, "Interaction Analysis: Foundations and Practice," *J. Learn. Sci.*, vol. 4, no. 1, pp. 39–103, Jan. 1995, doi: 10.1207/s15327809jls0401_2.

[13] K. P. Beier, "UM-VRL: Virtual Reality: A Short Introduction," 2004. http://www.umich.edu/~vrl/intro/ (accessed May 20, 2020).

[14] R. D. Brown, "Welcome to the world of virtual reality," 2001. .

[15] N. Negroponte, *Being digital*, 1st Vintag. New York: Vintage Books, 1995.

[16] M. Slater and M. Usoh, "Presence in immersive virtual environments," *Proceedings of IEEE Virtual Reality Annual International Symposium*. IEEE, doi: 10.1109/vrais.1993.380793.

[17] J. Boaler, L. Chen, C. Williams, and M. Cordero, "Seeing as Understanding: The Importance of Visual Mathematics for our Brain and Learning," *J. Appl. Comput. Math.*, vol. 05, no. 05, 2016, doi: 10.4172/2168-9679.1000325.

[18] D. C. D. Pocock, "Sight and knowledge," *Trans. Inst. Br. Geogr.*, pp. 385–393, 1981.
[19] J. T. Christopherson, "The growing need for visual literary at the university. VisinQuest: Journeys toward Visual Literacy." In 28th Annual Conference of the International Visual Literacy Association, Cheyenne, WY'de sunulmuştur, 1997.
[20] E. B. Kleinman and F. M. Dwyer, "Analysis of Computerized Visual Skills: Relationships to Intellectual Skills and Achievement," *Int. J. Instr. Media*, vol. 26, no. 1, pp. 53–70, 1999.
[21] R. Heinich, *Instructional media and technologies for learning*, 5th ed. Englewood Cliffs, N.J.: Merrill, 1996.
[22] Statista, "Age breakdown of video game players," 2019. https://www.statista.com/statistics/189582/age-of-us-video-game-players-since-2010/ (accessed May 20, 2020).
[23] R. Moreno and R. Mayer, "Interactive Multimodal Learning Environments," *Educ. Psychol. Rev.*, vol. 19, no. 3, pp. 309–326, 2007, doi: 10.1007/s10648-007-9047-2.
[24] M. P. Driscoll, *Psychology of learning for instruction*, 3rd ed. Boston SE - xix, 476 pages : illustrations ; 24 cm: Pearson Allyn and Bacon, 2005.
[25] A. Holzinger, M. Kickmeier-Rust, and D. Albert, "Dynamic Media in Computer Science Education; Content Complexity and Learning Performance: Is Less More?," *Educ. Technol. Soc.*, vol. 11, no. 1, pp. 279–290, 2008.
[26] E. Tse *et al.*, "Child computer interaction: workshop on UI technologies and educational pedagogy," in *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, 2011, pp. 2445–2448.
[27] J. M. Wing, "Computational thinking," *Commun. ACM*, vol. 49, no. 3, pp. 33–35, 2006.
[28] R. A. Duschl, H. A. Schweingruber, A. W. Shouse, and National Research Council, *Taking science to school: Learning and teaching science in grades K-8*. Washington, DC: National Academies Press, 2007.
[29] D. Weintrop *et al.*, "Defining computational thinking for mathematics and science classrooms," *J. Sci. Educ. Technol.*, vol. 25, no. 1, pp. 127–147, 2016.