Integrating Augmented Reality and Collaborative Activities to Enhance Computational Thinking in K-12 Classrooms

Seo, M., Kwon, K., Thomas, B., Kim, H., & Kim, K.

Abstract: This study explores the integration of Augmented Reality (AR) and collaborative activities to leverage abstract Computational Thinking (CT) concepts accessible to young students. The instructional design follows Plan, Act, Reflect (PAR) cycles that consist of three types of collaborative activities: Hands-on, AR-integrated, and self-directed robot programming activities. Findings highlight the importance of scaffolding in helping young learners, particularly those with low spatial ability, grasp directional concepts. Role-based collaboration proved effective in fostering engagement and problem-solving skills, though challenges emerged in the AR-based activity. This study contributes to immersive learning by demonstrating practical application of AR technology into K-12 classrooms.

Background and purpose

Despite the fact that computational thinking (CT) has become an increasingly important problem-solving approach in K-12 education, its abstract nature can present a challenge, particularly for young students (Tang et al., 2020). The embodied learning approach connects these abstract concepts to concrete experiences (Kopcha & Ocak, 2023). Augmented reality (AR) immerses students in embodied activities within an interactive digital environment (Lindgren & Johnson-Glenberg, 2013). Dunleavy and Dede (2013) explain that an AR experience on mobile devices can overlay data, simulations, and videos onto real-world scenes viewed through the device's camera. However, there is a gap between the characteristics of AR technology and its application. AR experiences are often designed for a single user and rely heavily on personal computing devices, such as smartphones or tablets (Dirin & Laine, 2018). And this creates two significant challenges. First, a single-user system can limit collaboration (Yuill & Rogers, 2012). In traditional classroom environments, students are encouraged to work together, sharing ideas and discussing solutions. But with AR, if each student is isolated on their device, it can restrict their interaction, reducing the opportunities for peer learning and collaboration. Secondly, the sole reliance on personal computing devices presents logistical challenges (Varier et al., 2017). Not all students may have access to the necessary technology, and managing a large number of individual devices in a classroom setting can also be complex for teachers. In this regard, this study explores an instructional design that can engage students in interaction with not only the mixed-learning application but also each other, thereby expanding their learning process and fostering a collaborative and inclusive learning environment.

Methods

This study is centered around an AR application designed to facilitate the practice of CT concepts through embodied learning. The application is operated by a user's four directional movements, which encompass moving forward, backward, left, and right. In essence, these movements are equivalent to coding instructions for constructing algorithms. The primary tasks within the AR application involve navigating from a starting point to an endpoint on a grid, while avoiding obstacles and completing missions. Users develop their own paths within four consecutive tasks in a single scenario. Demonstrations by the teacher and on-demand support from researchers were provided.

Instructional design

Integrating immersive technology such as AR into learning experiences requires a pedagogical approach that aligns intended learning objectives with learning activities, while considering design elements related to immersive experience (Wagner & Liu, 2021). Dede et al. (2017) emphasize the importance of the Plan, Act, Reflect (PAR) cycle in immersive learning experience for mastering complex knowledge and skills. In this model, students first prepare for an experience (Plan), then engage in the activity (Act), and finally evaluate their performance, identifying successes, areas for improvement, and necessary adjustments (Reflect). While immersive learning is particularly effective in the Act phase, it must be carefully designed to support the Plan and Reflect stages as well. These considerations highlighted the instructional design strategies for integrating an immersive learning environment.

The instructional design includes three collaborative pathfinding activities aimed at enhancing students' understanding of computational thinking (CT) concepts, particularly coding and sequencing. These activities are designed to be engaging, interactive, and conducive to active learning.

The first activity is a hands-on role-playing exercise that helps students grasp directional symbols as codes and paths as sequences of multiple codes. In groups of four to five, students interact on a 5×5 checkerboard mat, using their bodies to simulate programming. One student acts as a bee, another gives directional commands using symbol cards, while the remaining members set up mission points, including start and endpoints, obstacles, and targets. Students rotate roles to experience different aspects of programming. This activity follows a teacher-led demonstration using body movements to reinforce directional concepts.

The second activity is an AR-integrated paired exercise designed to foster collaboration in pathfinding (sequencing) and debriefing (debugging and optimization). In pairs, students engage in a role-playing game on a larger 5×5 grid. One student, the pilot, navigates the AR application, while the other, the documenter, observes and records movements. Students switch roles to ensure balanced participation. After each programming session, a debriefing phase requires students to organize directional symbols based on the documenter's worksheet, reinforcing their learning. Researchers facilitated this activity to guide implementation.

The final activity is a collaborative robot programming task that allows students to apply sequencing and debugging skills. Working in groups of four to five, students select a starting point and destination from a situational storyboard featuring a 5×6 grid. They then program a BeeBot to navigate the path, solving the problem collaboratively. This self-directed activity encourages students to integrate and apply their CT knowledge in a practical, hands-on manner. Table 1 demonstrates the alignment of three phases of instructional design with the PAR cycle and CT objectives.

Table 1

Three Phases of Instructional Design with PAR and Computational Thinking Objectives

Activity	**Plan**	**Act**	**Reflect**
Hands-on Activity	• The teacher lectured with body demonstrations explaining directional symbols, codes, and sequences • Introduction to role-playing activities and collaboration rules	• Use directional symbol cards to program a classmate's movements • Take turns playing different roles to experience sequencing	• Discuss how directional symbols correspond to codes • Reflect on the challenges faced while sequencing movements
AR- integrated paired activity	• Explanation of programming concepts such as sequencing and debugging • Introduction to operating an AR through a device for navigating the immersive environment	• One student (pilot) moves in the AR space, while the other (documenter) records sequences • Students switch roles and debrief using directional symbol cards and worksheets	• Analyze errors using the documenter's worksheet • Discuss optimization strategies for finding the best path
Robot programming activity	• Introduction to operating a robot for programming • Assigning students to groups or pairs for structured collaboration	• Use a Beebot to navigate through the grid • Solve problems by planning, executing, and debugging robot movements.	• Evaluate the effectiveness of navigation strategies • Modify and debug sequences to improve efficiency

Corresponding CT objectives	• Understanding directional symbols as fundamental programming concepts • Decomposing a navigation problem into smaller, programmable steps • Collaborating and assigning roles for structured problem-solving	• Applying sequencing to create a logical path. • Using abstraction to focus on key elements of navigation while ignoring distractions • Debugging errors by identifying incorrect moves and adjusting commands	• Evaluating the efficiency of programmed sequences • Identifying and correcting logical errors in the code • Engaging in iterative problem-solving through trial and error • Refining strategies for improved problem-solving

Implementation

We implemented the instructional design into a public elementary school located in the Midwestern United States. The participants were 44 students from 1st (n=17) and 2nd-grade (n=27). The implementation was conducted over a span of 5 days for each grade level to keep students' awareness and understanding of this instruction. It followed the sequence: teachers' demonstrations (approximately 10 minutes), hands-on group activity (approximately 40 minutes), AR-integrated paired activity (approximately 30 minutes per pair), and robot-programming group activity (approximately 40 minutes). Every activity was facilitated by researchers and teachers. In adherence to ethical guidelines, parental or guardian consent was diligently obtained, affirming the participants' voluntary involvement in this study.

Findings

The implementation of the instructional design revealed three key findings. First, structured teacher demonstrations play a crucial role in supporting the initial understanding of CT concepts, particularly for younger students in 1st and 2nd grade. At this developmental stage, students often struggle with spatial orientation, such as distinguishing left from right. Teacher demonstrations help bridge this gap by providing concrete cues. For instance, one effective scaffolding method involved using hand gestures, such as instructing students to form an "L" shape with their left hand to easily recognize directions. This structured guidance establishes a strong foundation for associating directional concepts with embodied symbols, ensuring a smoother transition into subsequent activities.

Second, the hands-on group activity proved highly effective in engaging students with CT learning by assuming different roles. Students became actively involved in the learning process, whether by guiding a classmate's movement, collaboratively solving tasks, or demonstrating resilience in overcoming errors. This role-based approach fosters teamwork and deeper conceptual understanding. Additionally, just-in-time facilitation played a crucial role in maintaining the activities' effectiveness, ensuring that students remained on track and engaged.

Third, integrating a single-user AR application within collaborative learning activities presents significant potential but also introduces notable challenges. The role of the documenter proved particularly challenging despite several aids, such as in-app auditory cues and simultaneous facilitation, to help documenters follow the pilot's movements more effectively. Many documenters struggled to track the pilot's movements, especially when the pilot moved too quickly or took complex, exploratory paths. This led to confusion, frustration, and difficulty in accurately recording movement sequences. Furthermore, some documenters disengaged from observing the pilot and instead attempted to solve tasks independently using the worksheet, resulting in a fragmented learning experience rather than a truly collaborative one. Addressing these challenges is critical for promoting meaningful and effective collaboration in the paired activity.

Discussion

The findings from this study have several implications for instructional design when integrating AR and collaborative activities to enhance CT in K-12 classrooms. First, scaffolding is essential throughout the learning process. As students progress through different instructional phases–beginning with teacher demonstrations, followed by hands-on activities, AR-based practices, and culminating in self-directed collaborative tasks–the level of instructional support should be gradually reduced. However, the teacher's role remains critical, particularly for younger learners struggling with spatial reasoning. Providing structured guidance at the outset and then gradually shifting responsibility to students ensures a more effective and independent learning process.

Second, the instructional design should ensure cohesive learning experiences that interconnect computational concepts, participant roles, and instructional strategies. Consistent exposure to key CT principles across diverse activities allows students to construct algorithms, develop problem-solving skills, and gain experiential intuitions about computational operations. Revisiting concepts through varied sensory interactions reinforces understanding and prepares students for future learning (Dede et al., 1999). The iterative nature of these activities fosters deeper comprehension and retention of CT concepts.

However, the challenges observed in AR-integrated activity, particularly in collaborative dynamics, highlight areas for improvement. The documenter role, in particular, requires additional support to prevent disengagement and improve performance in tracking the pilot's movements. Several strategies could enhance this role. For example, structuring the documenter's task into several part-tasks with designated stopping points where the pilot pauses would help students to deal with its complexity by allowing the documenter to verify and confirm their records.

Future research should further explore how AR can be optimized for collaborative learning in CT frameworks. Enhancing AR's ability to foster meaningful peer interaction and shared problem-solving experiences will be key to maximizing its educational potential. By addressing the observed challenges and refining collaborative elements, the instructional design can better leverage AR's strengths to support CT education.

References

Dede, C., Salzman, M., Loftin, B., & Sprague, D. (1999). Multisensory immersion as a modeling environment for learning complex scientific concepts. In W. Feurzeig & N. Roberts (Eds.), Computer Modeling and Simulation in Science Education (pp. 282–319). Springer.

Dede, C.J., Jacobson, J., Richards, J. (2017). Introduction: Virtual, Augmented, and Mixed Realities in Education. In: Liu, D., Dede, C., Huang, R., Richards, J. (eds) Virtual, Augmented, and Mixed Realities in Education. Springer, https://doi.org/10.1007/978-981-10-5490-7_1

Dirin, A., & Laine, T. H. (2018). User experience in mobile augmented reality: emotions, challenges, opportunities and best practices. Computers, 7(2), 33. https://doi.org/10.3390/computers7020033

Dunleavy, M., & Dede, C. (2013). Augmented reality teaching and learning. In J. M. Spector, M. D. Merrill, J. Elen, & M. J. Bishop (Eds.), Handbook of Research for Educational Communications and Technology (4th ed., pp. 735–745). Springer.

Kopcha, T. J., & Ocak, C. (2023). Children's computational thinking as the development of a possibility space. Computers and Education Open, 5, 100156. https://doi.org/10.1016/j.caeo.2023.100156

Lee, E. A. L., & Wong, K. W. (2014). Learning with desktop virtual reality: Low spatial ability learners are more positively affected. Computers & Education, 79, 49-58.

Lindgren, R., & Johnson-Glenberg, M. (2013). Emboldened by embodiment: Six precepts for research on embodied learning and mixed reality. Educational Researcher, 42(8), 445-452.

Tang, X., Yin, Y., Lin, Q., Hadad, R., & Zhai, X. (2020). Assessing computational thinking: A systematic review of empirical studies. Computers & Education, 148, 103798. https://doi.org/10.1016/j.compedu.2019.103798

Varier, D., Dumke, E. K., Abrams, L. M., Conklin, S. B., Barnes, J. S., & Hoover, N. R. (2017). Potential of one-to-one technologies in the classroom. Educational Technology Research and Development, 65, 967-992. https://doi.org/10.1007/s11423-017-9509-2

Wagner, C., Liu, L. (2021). Creating immersive learning experiences: A pedagogical design perspective. In: Hui, A., Wagner, C. (eds) Creative and Collaborative Learning through Immersion: Interdisciplinary and International Perspectives. Springer, https://doi.org/10.1007/978-3-030-72216-6_5

Yuill, N., & Rogers, Y. (2012). Mechanisms for collaboration: A design and evaluation framework for multi-user interfaces. ACM Transactions on Computer-Human Interaction (TOCHI), 19(1), 1-25.

Acknowledgments

This research is based upon work supported by the National Science Foundation (NSF) under grant no. DRL-2048989, ‘Supporting Early Learning of Computational Thinking Using Mixed Reality Technology’.