Supporting Children as They Program to Make Physical and Virtual Objects Interact

Amon Millner
Massachusetts Institute of Technology, Media Laboratory
77 Massachusetts Ave E15-020F
Cambridge, MA 02139
millner@media.mit.edu

Abstract

I develop technological tools to help communities of children creatively solve problems as they program computers. The Hook-ups project introduces computational tools that support children in building physical input devices for computer programs they create. Research questions guiding my work are presented, as are the methods by which I investigate them in informal learning environments.

Keywords

Design, prototyping, programming, input devices

ACM Classification Keywords

H.5.2 [User interfaces]: interaction styles, prototyping

Introduction

Today's 8-year-olds are likely to grow up in a technology-rich reactive environment. For example, they walk through doors that open automatically and play with toys that talk or wiggle when touched. As computational technology continues to play an increasing role in our biological, natural, and engineered environments, how one can fully participate in society is changing. Greater levels of technological fluency will be required to capitalize on these changes.

Objectives

My Ph.D. project focuses on helping young people develop understandings of computation that will give them options for how they live and work. I develop and explore sensing devices and programming languages to meet the interests, needs, and capabilities of children. I aim to make tools engaging and inspire children to take on diverse challenges presented by building things that combine physical materials and virtual media. My three main goals for the project are to: (1) develop technologies based on educational theories that enable a new range of design activities that combine physical and virtual design processes; (2) design strategies for introducing a new system to learners in informal-learning environments; and (3) deepen our understanding of how computational tools can inspire creative problem solving by youth.

In order to achieve these goals, I must build upon appropriate educational theories. An effective learning system combines complementary educational philosophies, tools embodying those philosophies, and settings in which the tools can be used. Papert's constructionist theory of learning suggests that young people learn best through the process of constructing artifacts [4]. My work combines Papert's notion of "microworlds" (simplified computer environments created to highlight a set of concepts) with his early interest in utilizing physical objects to introduce children to programming (as demonstrated with the floor-turtle project). In 2003, Eisenberg noticed researchers designing unique computer-based microworlds [2, 5] yet couldn't identify many who blended physical and virtual activities to improve the types of computer-enhanced educational experiences Papert envisioned [3]. Eisenberg then researched creative ways for computer programs to act on the physical world through fabricating physical objects [3], while other researchers developed programmable construction kits [7]. In contrast, the "Hook-ups system" I developed offers youth tools and activities that enable them to make physical objects interact with virtual media. This makes it possible for children to find new entry points to and alternative pathways through programming and physical design activities.

Making tools that support the integration of disparate types of media offers opportunities for people with different competencies to learn from one another in groups. There are points in group work in which assistance helps a learner further develop his or her understandings. In these zones of proximal development [8], the level of understanding to which someone arrives through building alone is less than the level of understanding he or she could reach with help from someone or something. I develop Hook-ups tools conducive to working in groups so that such development can be realized.

When projects warrant group work, it is possible that individual members' enjoyment and engagement can benefit, according to Csikszentmihalyi's concept of Flow [1]. He argues that groups of people can work together in ways that allow individuals to achieve Flow states. Some components for achieving such states are: clear goals, concentration on the task, loss of self-consciousness, direct feedback, balance between ability level & challenge, sense of control, an altered sense of time, and awareness of effortlessness in actions. Some contributing factors to facilitating Flow among individuals in group settings are individual differences and spatial arrangement - two factors of concern in my research project as well.

Research Questions and Methods

The research questions that guide how I develop tools, activities, and materials to promote children programming creatively are:

Design Question

1. What types of tools can be designed to facilitate novices building and sharing ideas for making physical and virtual objects interact?

Engagement Question

2. What types of activity structures provide novices with entry points into design processes capable of sustaining youth engagement in making physical and virtual objects interact?

Learning Question

3. As youth engage in, troubleshoot, use, and reflect upon design processes that make physical and virtual objects interact, what design concepts and programming strategies and do they learn?

The results that these research questions yield are likely to vary based on the backgrounds of the children using the tools. My choice to introduce this work to community technology centers (CTCs) in low-income areas while investigating the questions will influence which methods will inform my project. I will collect data from multiple sources during activities. I will use software logs, pre- and post- activity informal interviews, online surveys, screen recording software, artifact analysis, protected user blogs, audio/video recordings, and notes from participant-observers to inform my study analysis. I intimately understand challenges to evaluating designs that become integrated into unique learning spaces. It is difficult to tease apart factors that influence outcomes. I will consider such difficulties as a part of my Ph.D. work.

I will use the case study method to present how activity structure influences projects. The structure of activities using tools I develop will range from more-structured to semi-structured to less-structured activity. It will also dictate the time for activities, suggested theme, builder grouping, and access to material.

Work Done to Date

I have developed the Hook-ups system: a set of tools, activities, and materials to achieve my research objectives. Youth aged 10-18 have made Hook-ups - projects to control games, animations, and other computer programs they've created. In typical Hook-ups projects, youth integrate: sensors, physical materials to which sensors attach, sensor-querying interface boards, and sensor-input-ready programs.

Types of Hook-ups: Basic, Repurposed, and Fabricated

In making Hook-ups, youth have worked with materials ranging from everyday objects (such as plasticware) to output from personal fabrication devices (such as laser-cutters). For example, a girl made yoke-inspired Hook-ups for airplane-flying programs she created. In making a "basic" Hook-up yoke, she sketched and cut out a yoke from cardboard. After she added buttons to the cut-out and plugged wires from each into a sensor-input interface board on her computer, she wrote a program that moved an animated airplane – one button to move upward, the other to move downward. She later explored using parts from an electronic-toy steering wheel to make a "repurposed" Hook-up. She took it apart and used wire to connect its buttons to the program she wrote for the first project. In a later session, she created a "fabrication"-style Hook-up by using a drawing program to design a yoke shape to cut from a plastic sheet in a laser-cutter. She implemented the tilt contact switch she'd seen in the toy by hanging a thin plastic strip on the yoke so that its copper-covered tip would touch copper tape on either edge of the yoke when tilted. These contacts caused a new program's rear-perspective plane to roll left or right.

Initial Empirical Studies

Hook-ups work has shown ways that youth explored design concepts as they built and programmed prototypes. For example, some learned to build complex things from simple parts. Two groups of youth started building Hook-ups based on a game called Skeeball by creating a program that responded to a ball rolling onto a switch sensor in a cup. The groups then programmed complex actions for multi-cup models.

Iteratively Developed Tools and Activities

Most Hook-ups were programmed in the Scratch programming environment [6], which allows learners to build procedures by snapping together graphical blocks. Scratch provides command blocks that allow users to control objects on the screen. These control and sensing blocks can be mapped to keyboard keys, mouse movements, or external sensors.

Hook-ups work has been carried out with a simple interface board that turns analog readings from sensors into signals Scratch can understand. For example, an input device with a temperature sensor attached could control when a programmable "color filter" is applied to change the color of an on-screen character's face. Scratch supports communication with Hook-up board prototypes as well as a newer Scratch Sensor Board that people from all over (museums, schools, homes etc.) have been ordering from Scratch's website. As my Ph.D. work progresses, I will iterate the Scratch Sensor Board design and develop online tools to help people share ideas and information about Hook-ups projects.

Expected Contributions

My Ph.D. work will contribute to multiple research areas related to designing technologies for youth. For educators, I report insights into how children learn as they build Hook-ups. For designers and developers, I offer a set of guidelines based on empirical work. My insights and guidelines can serve as catalysts for future designers, developers, and educators to create systems that will enable youth to learn in new ways through making physical and virtual objects interact.

References

[1] Csikszentmihalyi, M., *Flow: The Psychology of Optimal Experience*. (1991), NY: Harper Perennial.

[2] diSessa, A.A., *An Overview of Boxer.* Journal of Mathematical Behavior, (1991)(10):p. 3-15.

[3] Eisenberg, M., *Mindstuff Educational Technology Beyond the Computer.* Convergence, (2003)(9:2):p.29-53.

[4] Papert, S., *Situating Constructionism*, in *Constructionism*, I. Harel and S. Papert, Editors. (1991), Ablex Publishing: NJ.

[5] Repenning, A., *Agentsheets: A Tool for Building Domain-Oriented Visual Programming Environments*, in *INTERCHI '93, Conference on Human Factors in Computing Systems*. (1993), ACM Press: NY.p.142-143.

[6] Resnick, M., Y.B. Kafai, and J. Maeda, *(ITR proposal for) A Networked, Media-Rich Programming Environment to Enhance Technological Fluency at After-School Centers in Economically-Disadvantaged Communities*. (2003), NSF.

[7] Resnick, M., et al., *Programmable Bricks: Toys to Think With.* IBM Systems Journal, (1996).**35**(3-4):p.443-452.

[8] Wertsch, J., *Semiotic Mechanisms in Vygotsky's Genetic Law of Cultural Development*, in *Vygotsky and the Social Formation of Mind.* (1985), Harvard Press: MA.