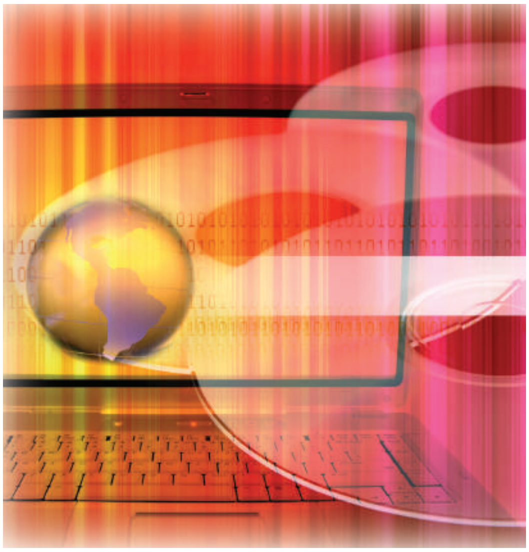

Kinecting to Mathematics through Embodied Interactions

Keri Johnson, Jebediah Pavleas, and Jack Chang,
University of Washington Bothell

Developed by a team of university students, the KinectMath software program encourages junior high and high school students to master abstract algebraic functions through embodied interaction using popular game-playing technology.

In a global economy increasingly dominated by computer technology, improving math and science education is a critical concern for schools across the US. Mastery of higher mathematics—beginning with algebra—is increasingly the gatekeeper for all STEM (science, technology, engineering, and mathematics)-related fields, academic and professional. Yet this is the point where students today begin to struggle and where traditional teaching methods often fail to motivate many of them to learn. To meet this need, new technological tools for the classroom, such as graphing calculators and dynamic graphing software, have emerged, and now Microsoft Kinect technology—originally developed for on-screen game playing using physical gestures as commands—provides opportunities to develop applications useful for teaching and learning algebraic and other mathematical concepts.

EMBODIED INTERACTION

Adapting Kinect technology to serve as a tool for teaching and learning ties into the emerging theory in cognitive science known as embodied cognition. This theory posits that intellectual understanding of concepts in the brain is enhanced by, and in some cases results directly or even exclusively from, multimodal bodily interactions with the physical world, or "embodied interactions." From the perspective of educators, this theory suggests that embodied interactions can be designed specifically to enhance learners' cognitive resources and serve as a foundation on which to build an understanding of abstract theoretical concepts. To this end, the idea of learning through embodied interaction can lead to technology-supported training activities that help students learn mathematical concepts through the experience, exploration, and practice of controlled bodily movements.

KinectMath software was designed for just these sorts of activities, where students can investigate and intellectually integrate mathematical concepts, such as linear and quadratic functions, by participating in physical performances that are replicated on-screen, just as they do when playing computer games.

THE KINECTMATH PROJECT

The KinectMath project originated with Robin Angotti, an associate professor of math education at the University of Washington Bothell and former high school math teacher, who wanted to help track students' distance over time and display the results in graph form using Microsoft Kinect. Existing software offered a standard graphing calculator and motion probe, but Angotti envisioned a more engaging and interactive user experience. Our team was eventually tasked with three overall goals for the project:

MEET THE DEVELOPMENT TEAM

KinectMath grew out of the combined efforts of a diverse development team; each member's unique perspective helped shape the project in significant ways.

Jeb Pavleas: While in high school, I was paralyzed from the chest down after a bad fall. I had always been passionate about electronic games, particularly the social and competitive aspects of game playing, but my disability barred me from many forms of play. Consequently, I began to adapt the controls of various gaming platforms, building them to suit my unique needs. Then, as an undergraduate in computer science at the University of Washington Bothell (UWB), I met Kelvin Sung through courses of his I took in game development and computer graphics. I approached him about working on a project that utilized a natural user interface via Microsoft's recently released Kinect sensor. He put me in touch with Robin Angotti, a professor of education who had an idea for an interactive math learning program. Based on her ideas, I built a quick prototype, and the project began to snowball from there. Jack Chang, a fellow student and friend, joined the project, and we began meeting weekly to incorporate the many ideas and requests that math teachers contributed to the program. As the project progressed, we continued to add features, modes, and options, and that process is ongoing. The whole experience solidified my interest in future game development and accessibility via technology.

Jack Chang: When I was a teenager, I spent all my time playing video games. For years, I didn't pay attention to anything else. I even dropped out of high school and never finished. Then a friend asked me, "If you like video games so much, why don't you learn how to make one?" He showed me the website of a UWB professor, Kelvin Sung. I was amazed by what I found there, so I decided immediately that I wanted to take his courses. I rushed and got my AA in a year and then applied and got into UWB. After taking Professor Sung's game development and 3D graphics courses, I was happy when he offered me the opportunity to work on projects with him—something I had waited a long time for. One of the projects was KinectMath. Jeb Pavleas, someone I worked with before in class, had already started the prototyping. In our first meeting, he showed me what he'd done in a week, while Professor Robin Angotti explained the whole idea of the project. What they were doing totally blew my mind. From that moment, I saw the great potential in the project, and I was eager to contribute and make it even better.

Keri Johnson: I joined this project when I entered the PhD program in secondary math education at the University of Washington, with a focus on technology in math education and how teachers can adopt technology in the classroom. Working as Robin Angotti's research assistant, I was initially involved in the design meetings for KinectMath, providing feedback to Jeb and Jack, along with other mathematics teachers, as they developed the software. Then I began working on curriculum materials to accompany the software and help teachers use KinectMath in their classrooms. I have been pleased because both teachers and students have been really excited to explore mathematics with KinectMath. In the future I'll continue to provide feedback to the developers based on teachers' classroom implementation, and I'll begin researching the impact this new tool has on students' mastery of mathematical concepts.

1. To help students understand abstract concepts by relating them to physical movements.
2. To demonstrate for students the connections between algorithmic equations and their graphical representation.
3. To engage students' interest, increasing their enjoyment and ultimately their understanding of mathematics.

Our combined efforts to create this embodied learning experience resulted in KinectMath, a practical software tool that allows students—and their teachers—to control abstract math concepts physically in real time by manipulating graphs and algebraic functions using their bodies, offering direct interactive involvement. (The sidebar "Meet the Development Team" provides additional information about KinectMath's development.)

Using the Kinect sensor's gesture recognition capabilities, KinectMath displays a user's movements via a familiar 2D graph onscreen, as shown in Figure 1. The software offers several modes of tracking and display from which to choose:

- *Tracking mode* tracks users' distance over time and reflects it back to them by plotting their position on the graph.
- *Matching mode* displays a random line on the graph and asks users to replicate the line based on their distance over time.
- *Editing mode* lets users easily manipulate graphs and functions by using their bodies. Based on simple hand gestures, users can change a line's slope, a parabola's width, or a sine wave's frequency and see those real-time changes reflected back in graph, equation, or table format.
- *Geometry* and *Bird's Eye modes* let users create geometric shapes and apply transformations by translating, scaling, and rotating them. The first projects these manipulations in frontal view, while the second shows the manipulations from a top-down perspective, similar to a map.

We developed these modes and the software's user interaction design based on regular weekly feedback from junior high and high school teachers testing various iterations of the software. This tight feedback loop encouraged new ideas and solutions to evolve naturally as the software matured. For example, our implementation of the relative fist-position controls demonstrated in Figure 1 grew directly from teacher feedback.

Stable versions of the software were eventually deployed in classroom settings, where we had the opportunity to evaluate the design's strengths and weaknesses, witnessing firsthand its use by both teachers and students. We believed this teacher-driven development would ensure the ultimate effectiveness of the KinectMath embodied learning experience.

KINECTMATH IN THE CLASSROOM

Initial implementations of KinectMath in high school mathematics classrooms have been highly positive. For example, students have used the Tracking and Matching modes to explore linear functions, specifically the concept of slope as a rate of change. In these modes, the software helps students understand, through embodied interaction, that the rate at which they move back and forth in front of the Kinect sensor creates a relatively steep- or flat-sloped line—steeper when their movement is fast, flatter when their movement is slow.

In Editing mode, students have used KinectMath to explore both linear and quadratic functions. For linear functions, students try to manipulate their own line to match a given line, for example "$y = 4x + 4$" as demonstrated in Figure 2. For quadratic functions, students attempt to create parabolas with given roots through their physical motion, while observing the factored form of the graph onscreen; this helps make the connection that the roots of the parabola are directly related to the factors of the quadratic function.

Instructors followed each KinectMath mode activity by having students answer a set of reflection questions about that activity before moving on to the next mode, after which the cycle would be repeated. Our observations reveal that students are generally excited by the process of manipulating algebraic functions using their bodies' motions and that they remain engaged and active throughout the lesson.

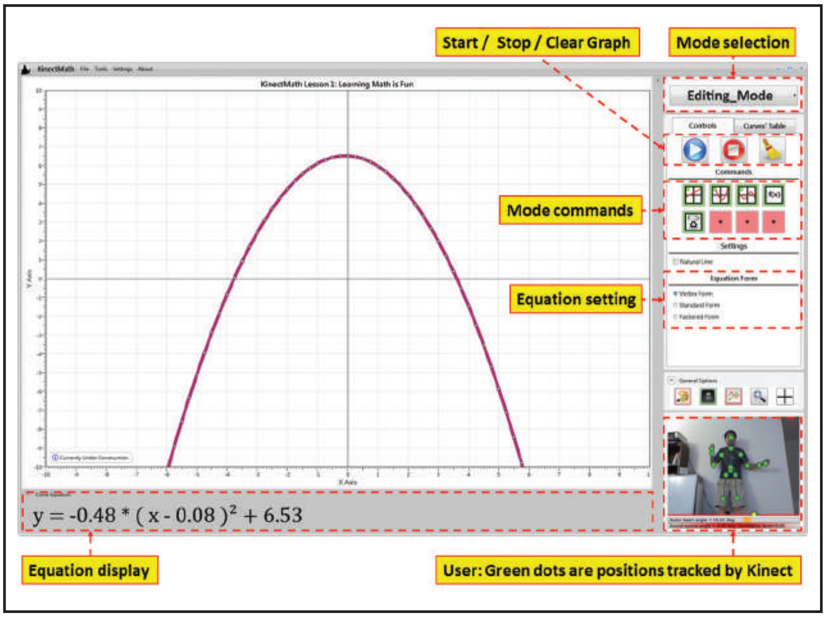

Figure 1. On-screen graph using KinectMath software. Editing mode lets the user manipulate a parabola. The relative position of the user's left fist controls the vertex position, while the horizontal distance between the two fists controls the parabola's curvature. (The user appears in the lower right corner of the screen.)

Figure 2. A high school student working in Editing mode with KinectMath. Here, the student is attempting to manipulate his left and right fists to match the line $y = 4x + 4$. The equation at the bottom left matches the line currently manipulated by the student's hand positions (the purple dotted line). The KinectMath software is running on the desktop computer to his right and projected on a SMART board.

KinectMath provides a concrete, motivating opportunity for students to explore abstract mathematical concepts through embodied learning experiences. Our future research will focus on other ways to implement KinectMath in the classroom and the impact the software has on specific student learning outcomes. As improving students' performance in algebra continues to be of critical importance, we hope that new tools such as KinectMath can contribute to the goal of motivating higher levels of classroom math achievement. ■

Keri Johnson *is a pursuing a doctoral degree in education at the University of Washington, with a focus on technology in secondary mathematics education. Contact her at kerij@uw.edu.*

Jebediah Pavleas *is a graduate student at the University of Washington Bothell majoring in computer science and software engineering. Contact him at jebp@uwb.edu.*

Jack Chang *is a graduate student at the University of Washington Bothell majoring in computer science and software engineering. Contact him at wei0831@uw.edu.*

Editor: Kelvin Sung, Computing and Software Systems, University of Washington Bothell; ksung@u.washington.ed. Professor Sung, with his colleague Robin Angotti, supervised the development of KinectMath.

Selected CS articles and columns are available for free at http://ComputingNow.computer.org.