

Enhancing Introductory Programming with Computer Game Technology

Gillian Smith

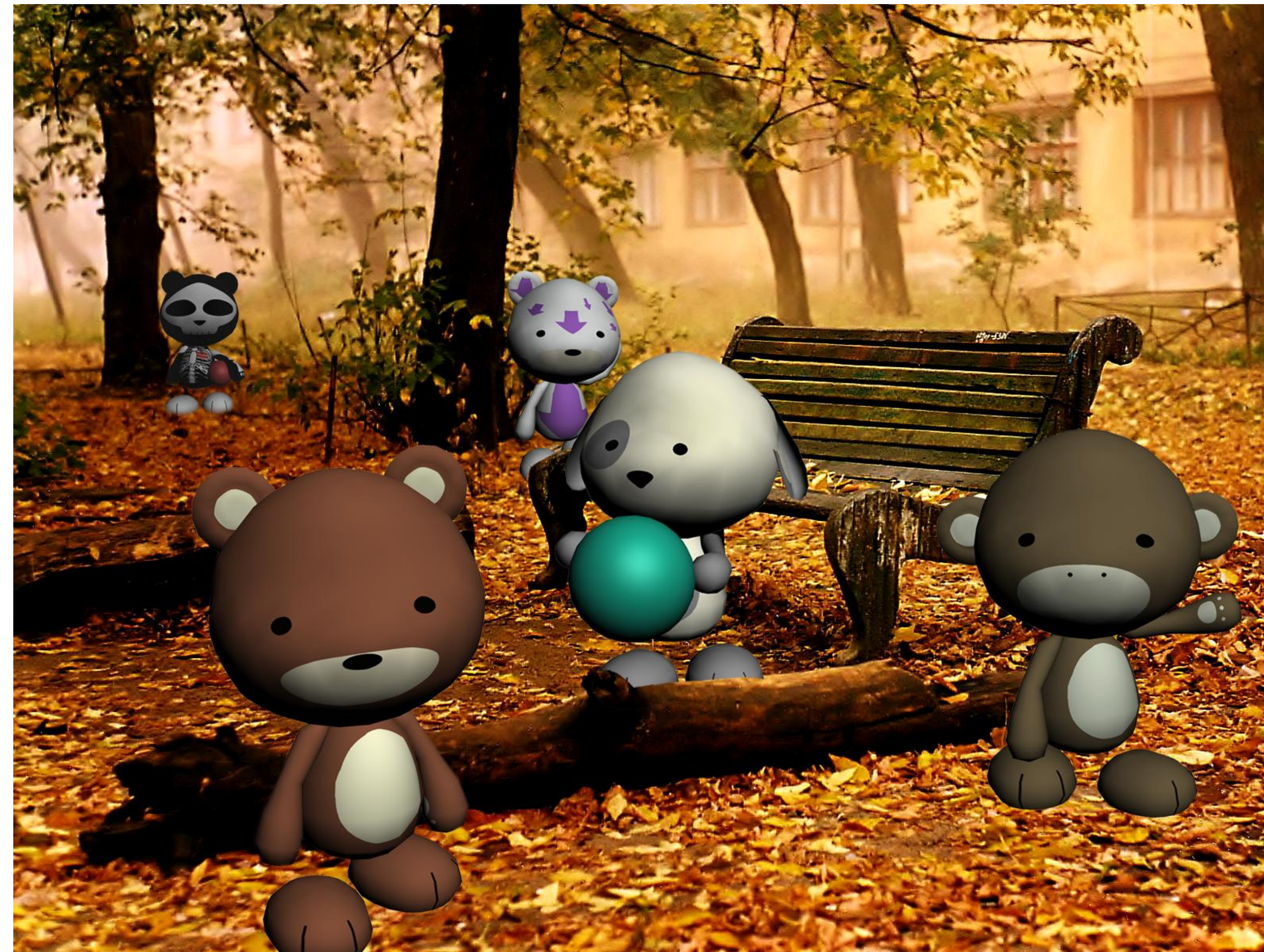


Jim Whitehead
Charlie McDowell

Objectives

- Use computer game technology to provide a more engaging environment for learning
- Teach basic programming concepts such as control flow and arrays with examples from computer games
- Encourage students to pursue Computer Science
- Increase interest in the Computer Science: Game Design major
- Current work-in-progress scheduled for use in Fall 2008

Concept Art



Example pet models, by Amber Okamura

Virtual Pets

- Provide early opportunities for students to personalize their work; current designs for pets include a dog, cat, mouse, and bear
- World contains a rich array of objects with which the pet can interact
- Pets are anthropomorphized; can express emotion through *The Sims*-style bubbles and facial expressions
- Allows a wide range of possible assignments; programmed behaviors can be simple or complex, depending on level of ability of student
- Provides an engaging environment for both genders

Teaching

- Relatively simple assignments provide interesting behaviors
- More advanced students are free to explore creating complex behaviors
- Three possible areas for students to do assignments:
 - Low-level programming of simple pet abilities, such as walking and jumping
 - Event-driven programming of more complex behaviors, such as fetching a ball or chasing a stick
 - Create a story involving the pet, with pet behaviors provided by the teaching staff
- MUPPETS provides built-in code editor and compile tool
- We provide all code for graphics so students can focus only on their assignments
- Non-competitive to encourage students of all abilities

Evaluation

- Students will be randomly split into two groups
 - Group A will be given programming assignments similar to those given in prior years
 - Group B will be given the Virtual Pet assignments
- All students have the same written homework assignments and take the same exams
- Comparisons can also be made with student performance from previous years

Acknowledgments

We thank Microsoft for funding our project. We also thank the Rochester Institute of Technology for their generous support of MUPPETS in its first use outside their program.