Helping Teachers and Students Learn to use 3D in AgentCubes Online

Catharine Brand
Science Discovery
University of Colorado, Boulder
Boulder, USA
catharine.brand@gmail.com

Alexander Repenning
Computer Science
University of Colorado, Boulder
Boulder, USA
Alexander.Repenning@colorado.edu

***Abstract*— We examine the challenges and advantages of 3D programming in the AgentCubes Online programming environment. 3D imagery and its associated programming provide a means for personalizing projects and may increase a learner's motivation to learn about technology. After reviewing the challenges for learners, we suggest some ideas for helping both teachers and student learners make the transition to 3D. We will present a plan for further research on this topic.**

Keywords—3D, game design, block languages

I. INTRODUCTION

Working with middle school teachers and students, the Scalable Game Design research group has investigated the advantages and challenges of block programming with condition and action blocks in if-then rules ordered in named methods. The project began with the 2D AgentSheets programming environment, then transitioned to the 3D AgentCubes downloadable program. Scalable Game Design teachers employed the 3D AgentCubes Online (ACO) programming environment during the 2016-17 academic year.

As we visited classrooms in 2017 to observe our SGD teachers in action, we noticed that many teachers utilized ACO to create single layer projects with 3D agents lying horizontally on the world plane and the same birds-eye view that they had used in the 2D AgentSheets program and did not take advantage of the more advanced 3D aspects of the new tool. Furthermore, most students could not or did not discover how to use the 3D options on their own.

Like creating their own agent representations, building their own 3D worlds would enable learners to personalize projects with very similar or identical programming. Our students have been very engaged in creating their own drawings for 3D agents. When interviewed, they express pride in the individual look of their ACO games and other projects. Taking advantage of the 3D aspects of ACO would build on this interest and possibly increase student motivation. In addition, creating programs that take full advantage of 3D is an intellectual challenge. Most of us can sketch in 2D but are challenged by 3D drawing. Is programming similar?

II. 3D FEATURES OF AGENTCUBES ONLINE

A. Agent Creation

The agents, which are the programmable characters and scene elements in ACO, can be drawn pixel by pixel on a 2D sketch pad. Specialized tools and options enable users to inflate a 2D drawing into a 3D image. The user also has the choice to stand the agent up so that the agent is visible when the world is rotated to a horizontal view rather than the 2D bird's-eye view.

B. World Building on a Single Layer

Teachers and students readily create ACO worlds that exist on a single plane. Since the world is most frequently viewed from the default birds-eye point of view, the 3D agents lie horizontally on the ground. We rarely see blocks or other agents piled on one another to create a structure or a multi-part agent. ACO does contain a programming block that allows a message to be sent to all of the agents in a stack so that stacked agents like a rider on a horse move together.

AgentCubes Online has tools that allow the students to rotate, pan and zoom their world so that they can tilt it to an almost horizontal position that allows upright 3D agents to be clearly viewed from the side while still showing the entire world. Most teachers and students do not use these tools unless they program first-person movement.

C. First-Person Movement

ACO allows a user to select an agent, click on the camera control and zoom in to see the ACO world from the agent's point of view. In order to make agent movement correspond to the first-person point of view, the movement programming must be augmented with methods that allow the agent in first-person mode to turn and then move in the direction it faces.

At our 2016 teacher professional development workshop, we taught sessions on implementing first-person movement. We provided a tutorial on first-person movement, with detailed explanations and exercises that helped learners build understanding about why the movement rules needed to be different in first-person mode. Experienced teachers picked up this concept and successfully taught it to their new students

978-1-5386-2480-7/17/$31.00 ©2017 IEEE

after they had created 4 or 5 programs or to their advanced students.

D. World Building on Multiple Layers

AgentCubes Online enables learners to add layers to the default single layer world. Each layer can support agents and many of the condition and actions blocks can be edited to refer to the layers by clicking on the triangle in the upper left corner of the block. For example, the basic move block shows where the agent should move. When the triangle is clicked in the corner is clicked, the hidden options appear and programmer may enter a positive number to tell the agent how many layers to move up or a negative number to specify how many layers down the agent will move. Layer 0 is the agent's current layer.

Teachers and students rarely use layers even though the buttons that create and remove layers are clearly labeled just to the right of the big central window. A slider controls the distance between layers and a checkbox makes the grid for a layer appear or disappear. Many learners may experiment with these controls. However, layers by themselves do not do much of interest. The user must program some agent behavior by un-hiding and using the layer specification options on the condition and action blocks and most students do not seem to notice the triangles or figure out what the layer specification means by themselves.

An additional challenge lies in imagining a project that uses layers in a rich, interesting way such as a 3D puzzle or a model of a 3D phenomenon like crystal formation.

III. Research questions

There are several different challenges to overcome in helping teachers and students use the 3D features of ACO. They break into distinct groups: interface issues, teacher training and developing curriculum to address the cognitive challenges of programming in three dimensions.

ACO has benefitted from years of testing in middle school classrooms with a single teacher and 15 to 30 students. The interface has been repeatedly redesigned to hit the sweet spot between brevity and information overload. A possible improvement would be to draw attention to the triangle on the blocks with hidden options by changing the tool tip as the mouse moves across these triangles.

However, teacher training and support will be the key to unlocking 3D in ACO for students. We saw a good response from teachers who attended the First-Person sessions in 2016. At our professional development workshop this summer, we added several sessions on 3D. These sessions began with a challenge to create a famous building on a single layer world, then went on to demonstrate how to program an agent to build structures on each layer of a multi-layer world, then finished with a recursive program to make a building duplicate itself on the layer above so that it looked like a skyscraper after several repetitions of the duplication. These sessions were well received as this comment from a teacher's post-session evaluation shows, "we did cover duplicating layers. Also awesome! I expect to use these features with my advanced kids.". We will survey teachers to discover how many intend to teach the 3D activities to students. We will also collect feedback on whether a single session is sufficient exposure for our teachers to feel confident about teaching this topic.

To support our teachers, we will produce curriculum on 3D topics for teachers as well as resources for students. The curriculum will begin with creating 3D agents for a single plane world. Then students will use the tilt, zoom and pan tools to turn the world so the viewer looks at it from the side. This set of lessons will also explore using relative location to direct an agent's sensing and movement by selecting one of the grid cells next to or on a layer above or below the agent. Students will learn to specify the layer based on how far above or below the agent it is located. Students will create a 3D game where a hero explores a treacherous environment while being pursued by villains.

The curriculum for advanced learners will also introduce 3D design methodology. Students will look at and analyze more sophisticated 3D games and models for inspiration. A suite of programs will demonstrate how to build a sophisticated 3D game step by step. Students will be encouraged to plan independent projects and to interleave drawing, building test 3D worlds in ACO and describing the agent behaviors before programming. Some learners might benefit from working on an actual 3D model made of Lego or cardboard, paper and strings to build intuition about agent behaviors in three dimensions. Building up understanding agent interaction can make it easier to figure out what methods and rules might be used to create the desired effects.

We also have an opportunity to work with an art teacher who will teach a semester-long advanced AgentCubes class to students who have already completed an introductory course. This teacher will test our 3D curriculum and work on 3D design strategies. We will be able to interview students to assess attitudes towards 3D in ACO and its impact on a sense of ownership and motivation to learn more about technology.

We believe that the use of 3D in projects can have an effect on student motivation similar to the effect of creating their own agent representations. Both of these activities allow students to personalize their projects, which increases a sense of ownership and as well as their motivation for learning more about technology**.** In addition, the exposure to designing and creating in 3D may enable some learners to discover new abilities that can be harnessed in future careers.

References

[1] A. Repenning, "Moving Beyond Syntax: Lessons from 20 Years of Blocks Programing in AgentSheets," Journal of Visual Languages and Sentient Systems, vol. 3, p. 24, 2017.

[2] Repenning, A., Webb, D. C., Brand, C., Gluck, F., Grover, R., Miller, S., et al., "Beyond Minecraft: Facilitating Computational Thinking through Modeling and Programming in 3D," IEEE Computer Graphics and Applications, vol. 34, pp. 68-71, May-June 2014.