Three to Four Dimensonial Brick Breaker

Matt Jordan

October 6, 2013

Introduction

For the longest time I have had no idea what to do my project on. I could think of no original ideas of my own so I dove into the archives of past projects to find something I could build upon. The first thing that caught my eye was the hyperbolic shooter from 2012, which made me think I wanted to do a game. The next project that caught my eye was 2D Snake, made in 2012 as well. It seemed like a natural step to make that game into 3 or 4 dimensions. However at the class cookout I found out that Vassyl was doing Snake as well so I wanted to do something original, but I liked the idea of taking a classic game and making it something more. Which brings me to my project idea: Three to Four Dimensional Brick Breaker. I will describe both potential projects and the initial problems I see with both of them.

Overview of 2-D brick breaker and what needs to be replicated-

In 2-D Brick Breaker one controls a paddle that slides across the x-axis in order to deflect a ball into bricks, breaking them. If the ball falls below the x-axis the player loses a life and is given a new ball to use. An interesting thing about the paddle is that, at least in the version of Brick Breaker I am most familiar with, it does not act like a flat panel when the ball hits it. If it acted like a flat panel the ball would hit the panel at an angle theta to the panel and then bounce off in the other direction at an angle theta to the panel. In Brick Breaker only the very center of the panel acts like this, the rest of the panel acts like it is tilted slightly in the negative y direction. This causes the ball to bounce in the direction of the side of the panel it hits. This is a feature I would like to replicate for my project.

In summary, I need to replicate:

1. A paddle sliding along an axis/plane/surface that deflects a ball

2. Detection for when the ball goes under said axis/plane/surface that will remove a life.

3. A paddle that does not deflect the ball perfectly, but instead deflects in a direction favoring the side the ball hits the paddle on.

How this would work in 3-D

In theory I do not think porting brick breaker to 3-D would be too hard. The paddle in the x-axis would now be a circle in the xy plane deflecting balls in the positive z direction. The player would lose a life if the ball falls below the xy plane and the paddle would act like a cone with a large angle at its vertex. My biggest problem with this version of the project is I don't know if the math component is different enough from 2-D brick breaker to make it a strong enough project. In my initial observations all the components move from 2-D to 3-D very easily and the math portion doesn't seem overly challenging. There are a couple of ways I could try to make this math component more prominent:

1. Have the ball move in non linear paths, possibly replicating air resistance in the xy direction.

2. Focus the math component on the generation of the bricks in interesting ways

3. Make a paddle that deflects the ball into interesting, non linear movements.

How this would work in 4-D

Making brick breaker in 4-D would be very interesting. The biggest challenge in this version is representing 4-dimensions in 3-space, and that is where the majority of my research thus far has gone. The most intuitive idea I've found in my research is that the user will manipulate data in 3-D, which the computer than translates into 4-D and after necessary computations projects the 4-D into 3-D for the user to see. This is a very broad interpretation of the concept and I still need to do some narrowing down of my focus. Some parts of the 4-D aspect of this project will be the same regardless. Firstly, the paddle would now have to be a sphere in the xyz space that deflects the ball into the fourth dimension, w. If the ball were to pass into the negative w region, the player would lose a life. The surface would have to act as though it were contorted in such a way that it favors deflecting the ball in the direction it hits the surface on, though I am once again not entirely sure what that would entail.

Challenges

Camera- Finding a workable camera angle for both 3-D and 4-D versions will be an interesting challenge. For 3-D a bottom-up view with a semi-transparent paddle is a possibility. For 4-D the camera will be heavily dependent on how I implement 4 dimensions in 3-space and will therefore will have to wait.

Non-standard deflection- Simulating non-standard deflection off the paddle will be a slight challenge, but should be overcome by designing a paddle with standard deflection and tweaking it.

The back-end

I will make this project in C++ using OpenGL. This allows me to take a language I already have some experience with, giving me a good starting point, and introducing myself to something I have never worked with before, giving me a useful tool to learn. Doing this in C++ with OpenGL is also beneficial because with Emscripten there is still a possibility that some stages of the project will be able to be used in HTML5 with Canvas as well.

Timeline

Week of 10/21/13 – Have a version of 2-D Brick Breaker ready to be shown in my presentation in class

Week of 11/04/13 – Have a working prototype of 3-D Brick Breaker

Week of 11/11/13 – Have the first stages of 4-D Brick Breaker done and a more complete 3-D version

Week of 11/25/13 – Have a working 4-D Brick Breaker program with only minor tweaking necessary

Week of 12/02/13 – Finish up documentation of project and make graphics easier on the eyes

Bibliography

Banks, David. *Interacting With Surfaces In Four Dimensions Using Computer Graphics*. Rep. no. TR93-011. Chapel HIll: n.p., 1993. *Univeristy of North Carolina Computer Science*. University of North Carolina. Web. 17 Oct. 2013. <http://www.cs.unc.edu/techreports/93-011.pdf>.