3D Boids Simulation

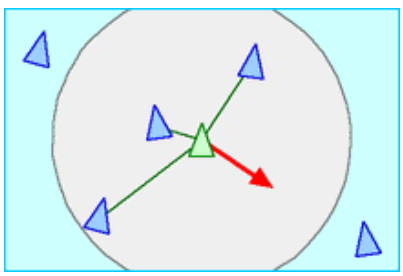
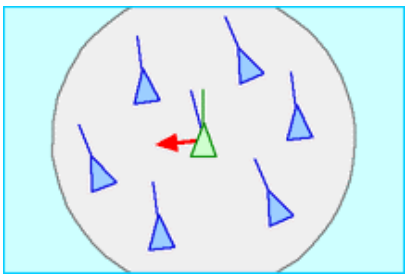
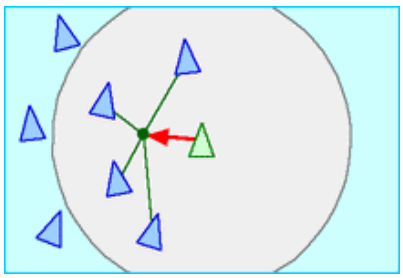
Paige Hinkle

Background

Boids is a program that simulates the behavior of birds when flying in a group. The original idea for the program was proposed by Craig Reynolds in 1986 in his paper “Flocks, Herds, and Schools: A Distributed Behavioral Model". The boids flock is similar to a particle system in which each particle or creature follows specific rules. The rules that Reynolds proposed are the following:

I. Separation: boids must not collide with other boids
II. Alignment: boids must attempt to match the velocity of nearby boids
III. Cohesion: boids must attempt to stay near the center of the flock

The three images below are from Reynold’s paper and illustrate separation, alignment, and cohesion respectively.

Implementation

Boids Model

Originally I planned to use simple tetrahedrons to represents the boids. However in the three.js examples there was a program that represented birds as triangles and included a wing animation based on a sin curve (http://threejs.org/examples/#canvas_geometry_birds). The code that I used from the example for the boid model is located in the obj/Bird.js file. From this example I also used the code to restrain the boids to a certain box in space. The example used a method that made the boids avoid the edges of the scene rather than just have them turn around and bounce off the edges of the scene as I had originally planned. This code is located in the function checkBounds().

Reynolds Model

I. ***Separation:*** boids must not collide with other boids

In order to create separation, I calculated the distance between the current boid and each of the other boids in the flock. If the distance between two boids is less than a certain separation factor then I calculate a vector that will push the boid in the opposite direction of the other boid. This vector is then normalized and weighted based on the distance between the two boids. These vectors are added up to solve for the final vector that will push the boid in away from other boids. The screenshots below show the effects of changing the separation factor. The image on the left has a high separation factor thus the boids are very spread out. The image on the left has a low separation factor thus the boids can flock closer together.

II. ***Alignment:*** boids must attempt to match the velocity of nearby boids

Alignment can be achieved by calculating the average velocity of the flock. The current boid's velocity is updated to be some percentage closer to the average velocity of the flock. This is calculated by subtracting the velocity of the flock from the velocity of the current boid and dividing by an alignment factor. Although velocity cannot be seen in a screenshot, you can see the effect of alignment in the direction of the boids below. The image on the right has a much lower alignment factor and you can see that the boids are pointing in the same direction. The image on the left contains boids that are not all pointing in exactly the same direction.

III. ***Cohesion:*** Boids must attempt to stay near the center of the flock

Similar to alignment, cohesion can be calculated by finding the position of the center of the flock by averaging all of the boids positions. Then the boids position is moved some percentage closer to the center of the flock. This is calculated by subtracting the center of the flock by the boid's position and dividing by some cohesion factor. In the images below you can see a high cohesion factor leading to a less cohesive flock on the right and a low cohesion factor leading to a more cohesive flock on the left.

Flockiness

In addition to separation, alignment, and cohesion I created a factor called flockiness. This factor controls what percentage of boids are subject to the separation, alignment, and cohesion factors. With a flockiness factor of zero, every single boid calculates its separation, alignment, and cohesion vectors every time the scene is rendered as seen in the right image below. With a flockiness factor of one hundred the boids are not subject to these factors, thus are just flying around with no regard for the flock as seen in the left image below.

Attraction and Repulsion

In this simulation the user is also able to attract or repulse the boids using the mouse. For both attraction and repulsion I calculated the distance from the target (the coordinates of the mouse) and the boid. If the boid is within a certain distance its velocity vector is directed towards or away from the target depending on whether the user is attracting or repulsing.

Perching
Boids also have the ability to perch. If a boid flies within a certain range of coordinates the boid will stop moving to simulate perching. The user can click anywhere to release the boids from their perch. In this simulation I made the perching coordinates align with the roofs of the first row of buildings in the background image however they could be set to the ground or the surface of another model.

Background Image
Originally I had planned on creating buildings as simple cubes for the boids to move around, however because of time I did not implement this. Instead I created a plane as the background texture with an image of Manhattan. This worked out perfectly because boids sounds like a person saying birds with a New York accent.

Improvements and Future Work
One bug in this program is when you change the separation, alignment, or cohesion factors drastically the flock oscillates as the boids try to find their new place in the flock. This could be smoothed out by slowing increasing and decreasing the factor.

Another bug that I would like to fix is having the option to include or exclude the perched boids in the calculations for separation, alignment, or cohesion. I thought it would be more natural for the boids to include their perched flockmates, but sometimes the flock just ends up flying in circles around the area where the other boids are perched.

There are many additional features that can be added to boids. One such addition would be adding a cityscape and having the boids fly around the buildings as previously mentioned. There could also be a feature that includes wind and suction or creating different flock shapes. There have also been simulations including different kinds of creatures such as bats, fish, and livestock.

Work Cited
http://www.red3d.com/cwr/boids/
http://www.kfish.org/boids/pseudocode.html
http://threejs.org/examples/#canvas_geometry_birds
http://www.cs.toronto.edu/~dt/siggraph97-course/cwr87/