**CS148 Final Project**

**Wenxin Zhou (wenxinzh)**

**Travel in Water Town**

Project proposal: I want to create a scene of a water town in China. A small boat traveling along the river, the camera follows the boat to explore the view alongside the river.

I plan to make two modes (day & night), if time permits. During the night, the lanterns outside the houses will light up and create reflections in the water.

**Inspiration:**

I want to recreate the scene in a more realistic setting.



Jiang Nan Shui Xiang by Yunpu Yuan

**MODEL CREATION**

I made all the 3D models in Maya from scratch and applied textures I found online.

Background: The background image was cropped from the original picture and blurred using GIMP. The background model was made by 6 planes.

House: The house model started out from a cube, then a pyramid was stretched into a triangular prism and attached to the top of the house. The roof and the wall have different textures applied to them. The window and door are two planes with image textures that are attached to the house model.

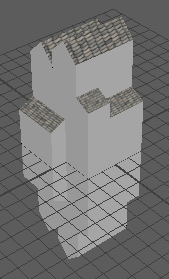
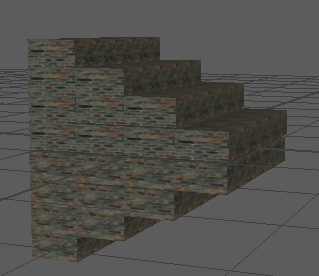
House complex: This model is made from scaling and combining several house models

Stairs: This model is made by putting 60 identical cubes together.

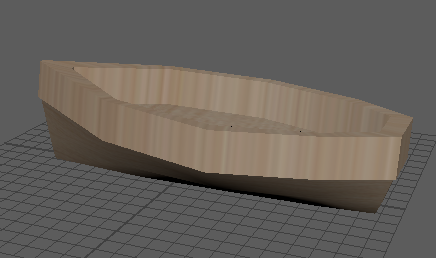
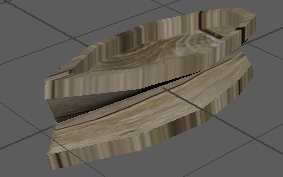
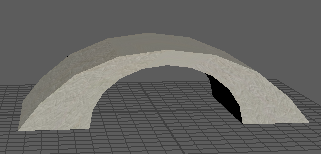
Bridge: The bridge was made by cutting and stretching a pipe. It deforms when it’s loaded into OpenGL hence it was left out of the actual simulation.

Boat: The boat was made by following an online tutorial.

All models have two parts and the bottom part was made by mirroring the top part. This was meant to represent the reflection in the water.

House Complex Stairs

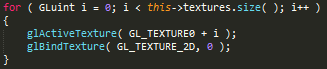
  

Boat Boat Bridge

**MODEL LOADING**

I used ASSIMP to load 3D models into OpenGL project as well as stb\_image.h to load the textures. I encountered a lot of problems in this part.

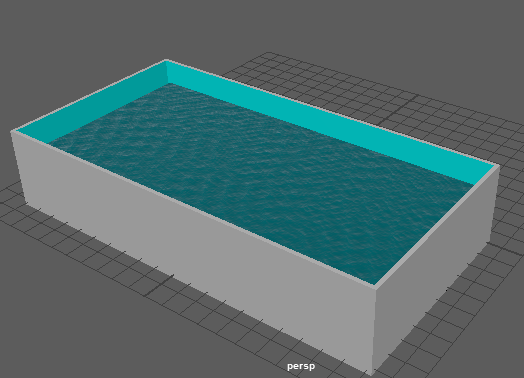
1. When I put two models in the scene, one textures overwrote the other. The reason was that I did not set the texture to default after finishing rendering.



1. The other problem was that the texture was tilted in the model. I had to cropped every image into a square.
2. Assimp did not load Maya default textures and material. I did not find a solution to this problem.

**WATER SURFACE**

I tried to simulate water ripple in Maya, I made the following model in Maya, however I did not figure out how to import Maya default material into OpenGL.



The ripple model in Maya

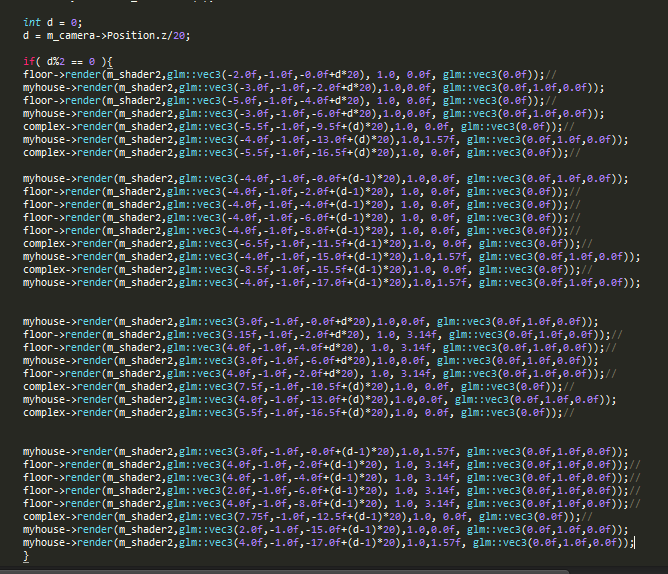
Then, I tried to just used a plane with water textures applied to it. I wanted the plane to be transparent but I did not know that I could adjust alpha channel in the shader to make it transparent. I drew a huge water surface plane with a bunch of triangle meshes, which was adapted from the Cube vertex data in hw2. I changed the shader so that it had 50% transparency.

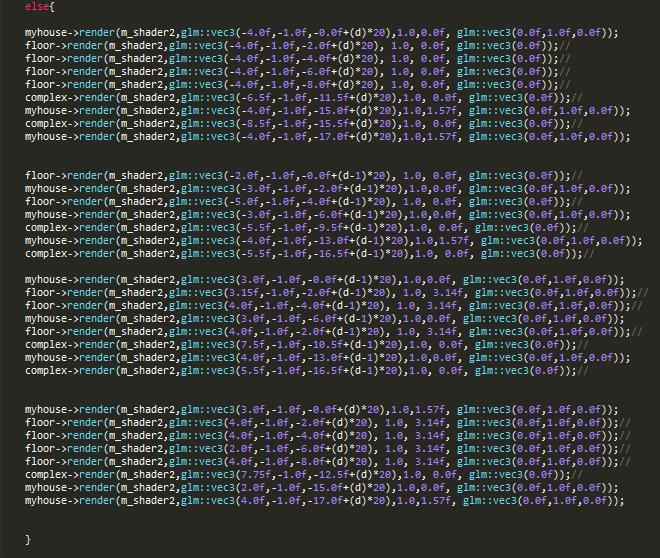
The water surface used two shaders, one is blue with 50% transparency. This is to represent the water in normal situation. The other white shader with 90% opacity is to represent the water surface after snow.

**POSITIONING OBJECTS**

The houses and other objects were generated procedurally as the camera was moving forward. In order to save the computational resources, I did not render all the objects at the same time, which was also impossible since the camera could go very deep. Therefore, I set 20 as one rendering distance. I rendered objects within twice of the rendering distance from the camera, which is area between the camera.position.z and camera.position.z + 40. To ensure the continuity of the scene, I structured the code such that the objects from camera.position.z+20 to camera.position.z + 40 were the same as the area camera.position.z+0 to camera.position.z + 20, when the camera moved to the next 20 region.

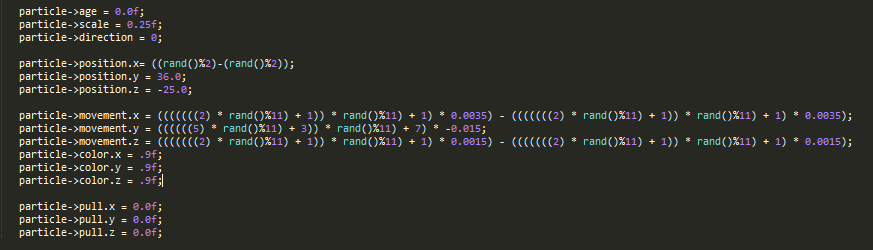




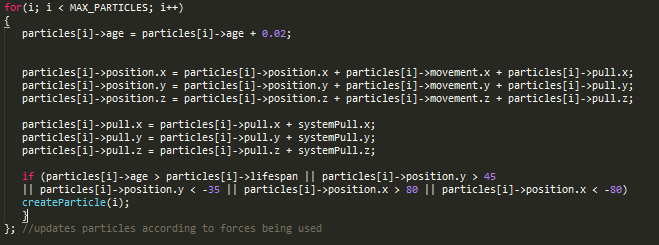
The models were rotated and positioned in the location carefully.

**PARTICLE SYSTEM**

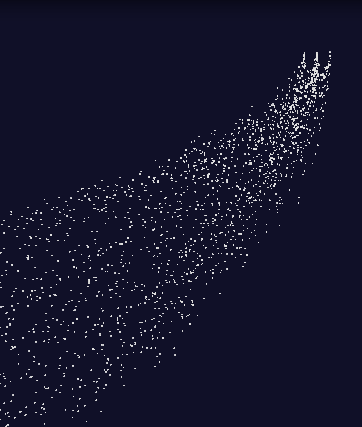
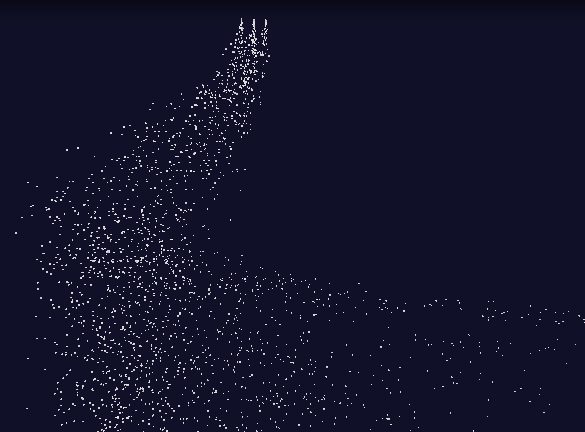
My original plan was to implement water ripple effect, due to the lack of time and research, I switched to simulate snow using a particle system. Each particle was rendered with a small cube, which was also adapted from Breathing Cube in hw2 code. The particle system is composed of 3000 particles. The initial positions of the particles were randomly generated with fixed y and z. movement.y is a negative number due to gravitational force. Pull represents the external forces such as wind. By pressing C or Z, we can change the external force to left or right.



In updateParticle method, new position of particle was calculated by adding current position, movement and external force. And new particles would be regenerated when the when the existing particles moveing out of range or existing longer than their lifespan.



No external force Wind to the left Wind to the left and then right





Link to demo: <https://youtu.be/3hGc6kqP4l4>

**Sources**

Painting

<http://yuanyunfu.artron.net/works_detail_brt000001600572>

Assimp

[https://learnopengl.com/#!Model-Loading/Assimp](https://learnopengl.com/)

<http://assimp.sourceforge.net/>

Model Loading

<https://github.com/SonarSystems/Modern-OpenGL-Tutorials/tree/master/%5BMODEL%20LOADING%5D>

Particle system

<http://www.swiftless.com/tutorials/opengl/particles.html>

Boat Model

<https://www.youtube.com/watch?v=5aI2nmC_TTs>

stb.h

<https://github.com/nothings/stb/blob/master/stb_image.h>