COMS 4160 Final Project
New Big City

John Fulgoni
Columbia University in the City of New York

14 May 2015

1 General Improvements

For my final project, I decided to add more features and expand upon my Programming Assignment 1, where I made a city where a character can walk around and explore a city grid.

1.1 Roads

To start, I simplified the road drawing process. In PA1, I made one city block, and then duplicated, translated, and rotated the buildings and roads in order to form a grid. In this version, I made the roads by simply drawing fewer 2D quads that would stretch across the entire city. I also textured the roads using simple texture of a double yellow-line road [18] that I found online.

1.2 Skybox

I added a skybox to my scene, which gives it a realistic look [3]. I found a code version online [4] that I had to edit, change, and adapt into my own class that could be drawn in my scene. I used a skybox image set found online [5], but had to create the image for the bottom of the box myself using image editing software. With this, I removed the "grass" that I had in PA1, so the user can look down and see the image of sand.

1.3 Lighting

I implemented simple lighting into my code using one position based light, and setting the normals for the buildings and people that I drew. This gives a more realistic image just based on having edges of the buildings, and the people reflect and block light.

1.4 Person Class

In PA1, the people walking around was based on one draw method that I would manipulate and animate through the rendering method. Now, the Person class exists by itself, which greatly reduced the code needed for animation. Each instance now contains a starting position, clothing color, and walk speed.

I also gave the main character the ability to jump using the left CTRL button, just using a simple method to increment and decrement the height of the character. There isn't anything to really "jump over", but I figured that I would leave the code in as a feature.

1.5 Game States

I added another scene to the city by using game states. By pressing the F2 button, the user can transport himself to the desert, where he'll see a Volkswagen parked next to some gravestones. The user can press F1 in order to go back to the city. The character will stay in the same global location upon switching states, so if you move the main character in the city, he will be moved in the desert. The desert and patches of the city also have the cacti [10] scattered across it.

2 New Models

I used a ton of new models that made it's way into the New Big City. These include cars [7, 8, 9], buildings [12, 14, 15, 16, 17], a helicopter [11], cacti [10], gravestones [13], and a snake [19]. I wanted to add these models to give the city a better look and feel to it.

2.1 Buildings

I added several new buildings to the scene. Instead of replacing my old buildings, I figured I would just expand the city in order to show before and after improvements. For the general small apartment building, I used the Raleigh Office model [12]. I made a display list to make one block of buildings, and then copied and translated the blocks to create the general city.

I also added several skyscrapers to the city. On the far east side, I used the Burj Arab model [14] that lies at the east end of the southern road. I also added copies of the Commerzbank building [15] that tower over the city from high above.

There are also two residential buildings that are in the new southern district [17], as well as two smaller buildings [16] that were designed to be pool complexes, but for the sake of my city could be anything.

2.2 Truck and Cop Car

I added two new moving ground vehicles, the truck [7] and the cop car [8]. Both of which have their own respective classes which dictate how they move. By using a series of if-else

statements and setting booleans, I make both sets of vehicles drive around different squares of the city on their own. From above, you can see the vehicles driving (near) the right side of the road, and some will even cross paths at certain times.

2.3 Helicopter

I also added a helicopter model [11] that flies in a circle above the city. To make the movement for the helicopter, I simply use take the sine and cosine of a rotation angle, multiply those by the desired radius, and set those values for the x and z.

2.4 Snake

In the desert game state (F2), I have snakes that slither around the map, back in forth in a line. I cropped the head of the snake from a turbosquid model [[19]] and used it as a segment in the body. The body is an altered version of the Forward Kinematics PA2 chain link, where I manually dictate what angle each joint has. I use the cosine function to give alternating values to the angles, which gives a wave like motion to the snake.

2.5 Shaders

In order to color the models that I imported, I used different versions of the shaders that I made for PA3. Some of these are no different from the ones submitted in the homework, while others are just basic modifications to color, or checker/stripe length. The shaders used for the cacti for example, is basically the Blinn-Phong shader from PA3.

3 Source Code Explanation and How to Run

Like PA1, the bulk of the code lies in BigCity.java. Any model that has animation has gotten its own class, while models like buildings are simply drawn from display lists. Skybox also has its own class that loads the textures and sets up the proper texture coordinates. My code is built on my submission of PA1, which is built on the starter code given my Professor Zheng for the first assignment.

OBJLoader.java, Model.java, and Face.java are all simple classes to load .obj files that I recreated from the video online [6]. Pokemon.java contains an earlier version of my project with all sides of buildings and characters color coded so that I could learn the orientation of the objects.

My code was developed using the Eclipse IDE. If ant is installed, the program should run with ant. A video is included with the source code, and there is also a longer video on YouTube [1] if you wish to see more of what I have done.

References

[1] Youtube: Longer Video for New Big City
https://youtu.be/hV0mXMTP3EI

[2] Fraps Real-time video capture software
http://www.fraps.com

[3] Cube Maps: Sky Boxes and Environment Mapping http://antongerdelan.net/opengl/cubemaps.html

[4] Simple Skybox https://code.google.com/p/lwjgl-water-shader/source/browse/trunk/src/edu/fhooe/mtd360/watershader/objects/SkyBox.java?r=27

[5] Red Sorceress, Skybox "Bleached" http://www.redsorceress.com/skybox.html

[6] YouTube: Oskar Veerhoek's LWJGL Tutorials - Video 24 3D Models
http://youtu.be/izKAvSV3qk0?list=PL19F2453814E0E315

[7] TurboSquid: Ford E100 Econoline http://www.turbosquid.com/FullPreview/Index.cfm/ID/717563

[8] TurboSquid: Mitsubishi Lancer http://www.turbosquid.com/FullPreview/Index.cfm/ID/541310

[9] TurboSquid: Volkswagen Type 2 http://www.turbosquid.com/FullPreview/Index.cfm/ID/874120

[10] TurboSquid: Cactus http://www.turbosquid.com/FullPreview/Index.cfm/ID/498238

[11] TurboSquid: Blackhawk Helicopter http://www.turbosquid.com/FullPreview/Index.cfm/ID/590210

[12] TurboSquid: Raleigh Office 03 http://www.turbosquid.com/FullPreview/Index.cfm/ID/655760

[13] TurboSquid: Grave http://www.turbosquid.com/FullPreview/Index.cfm/ID/312315

[14] TurboSquid: Burj Arab Medium Detail http://www.turbosquid.com/FullPreview/Index.cfm/ID/491073

[15] TurboSquid: Commerzbank http://www.turbosquid.com/FullPreview/Index.cfm/ID/722174

[16] TurboSquid: 3D Indoor Pool http://www.turbosquid.com/FullPreview/Index.cfm/ID/235808

[17] TurboSquid: Free Residential Building http://www.turbosquid.com/FullPreview/Index.cfm/ID/519486

[18] TurboSquid: Road Texture http://previewcf.turbosquid.com/Preview/2014/08/01__23_25_03/RoadTexture2.pngea1d5194-5e2b-4f81-8f89-19705d1f4fabLarge.jpg

[19] TurboSquid: Snake http://www.turbosquid.com/FullPreview/Index.cfm/ID/398555