Computer graphics report

Mike Hergaarden - MHN212
mike@M2H.nl / mhn212@few.vu.nl
15 January 2010

Overview

Having used 3D (game) engines such as Ogre3D (http://www.orge3d.org) and Unity (http://www.Unity3d.com), getting down and dirty with OpenGL was quite a challenge. My assignment has resulted in the tech demo as shown in the frontpage screenshot, with the following noticeable features:

- A desert scene (ground, western buildings)
- A cat walking in between two buildings
- A car driving in circles forever
- Multisampling
- Linear fog
- Skybox background
- Pause/Play control
- Camera movement (translation&rotation)

All of this has been created making use of a SceneGraph structure as showcased earlier in the 10th lecture of this course.
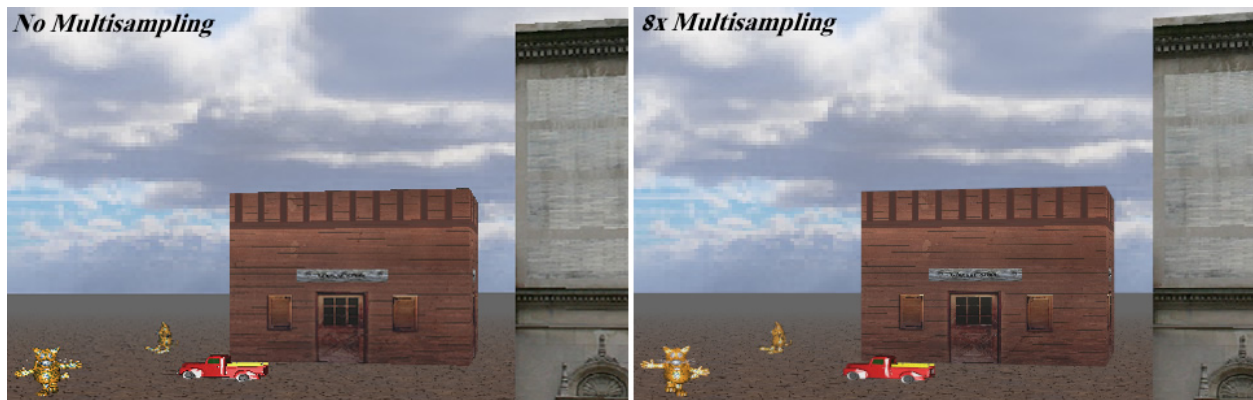
The most satisfying feature I've implemented was multisampling. While it is no technical breaktrough, to me it's feels like a must for every 3D application. It surprised me it took just a few lines to implement this in OpenGL. For the same reason I also added the skybox; I preferred creating a solid "OK"-looking base.

Design

The project uses the following two packages;

<u>Default package:</u>
- Controller.java
    - Can be used to control("animate") a Rotation and/or Translation. This is how the truck drives and how the cat in the background visits the two buildings.
- Project.java
    - The main application file. Loads all resources (models/textures) and initializes the SceneGraph. After all of this has been done the main loop (display) renders the skybox and scenegraph and processes the animation for the cat and ford. Input is processed on the "background" and applied whenever required.
- SGFLoader.java
    - The original SGFLoader file, with minor changes: Adds random texture coordinates when this option is set via the LoadModel method.
- Skybox.java
    - Used to setup and render the skybox.

<u>SceneGraph package:</u>
- AnyModel.java
    - Used to load&render a SGF format model. No fancy features.
- Camera.java - *(Lecture10)*
    - Camera abstract
- Cube.java
    - Renders a cube
- Geometry.java - *(Lecture10)*
    - Geometry abstract
- Light.java - *Lecture10)*
    - Represents/exposes an OpenGL light
- Material.java - *(Lecture10)*
    - Used to apply a material on SceneGraph children
- Node.java - *(Lecture10)*
    - SceneGraph node
- OrthographicCamera.java - *(Lecture10)*
    - SceneGraph implementation of an orthographic camera
- PerspectiveCamera.java - *(Lecture10)*
    - SceneGraph implementation of an perspective camera
- Player.java
    - This class represents a player and is used in this application for the Cat model. Since theres no real player in this application, with the current usage this class could've been renamed to Character. The player class animates the Cats body parts and renders them seperatly.
- Polygon.java
    - Class to render a polygon, used to render the desert ground plane.
- Rotation.java - *(Lecture10)*
    - For SceneGraph rotations
- Scaling.java - *(Lecture10)*
    - For SceneGraph scaling

- Seperator.java - *(Lecture10)*
  - SceneGraph 'dummy'
- Sphere.java - *(Lecture10)*
  - For rendering a sphere
- Texture.java
  - From lecture 12. Defines/loads textures.
- Transformation.java - *(Lecture10)*
  - Transformation abstract class
- Translation.java - *(Lecture10)*
  - For SceneGRaph translations

The models folder only contains the standard Computer Graphics course models. The ford and the cat model are used. The texture folder does contain some new additions; First there are the 6 skybox textures. The last three additions to the textures are the desert ground texture (desert.jpg) and the two house textures (house.jpg and westernhouse.jpg).

User manual

- Use the WASD keys to move the camera position (only moves in world space, regardless of cameras rotation).
- Look around by moving the mouse.
- Spacebar can be used to pause/play the application.
- Q or ESCAPE quits the application.