CSCI 3260 Principles of Computer Graphics

Course Project: Sim City (25%)

Demonstration Date: Apr 19 (Tuesday), 2016

No Late Submission is allowed

Fail the course if you copy

I. Introduction

In this project, you are given a preliminary town scene and the task is to use OpenGL to enrich it. The initial scene is as follows. Buildings, trees, roads, and vehicles are already provided, but more can be done to beautify them.

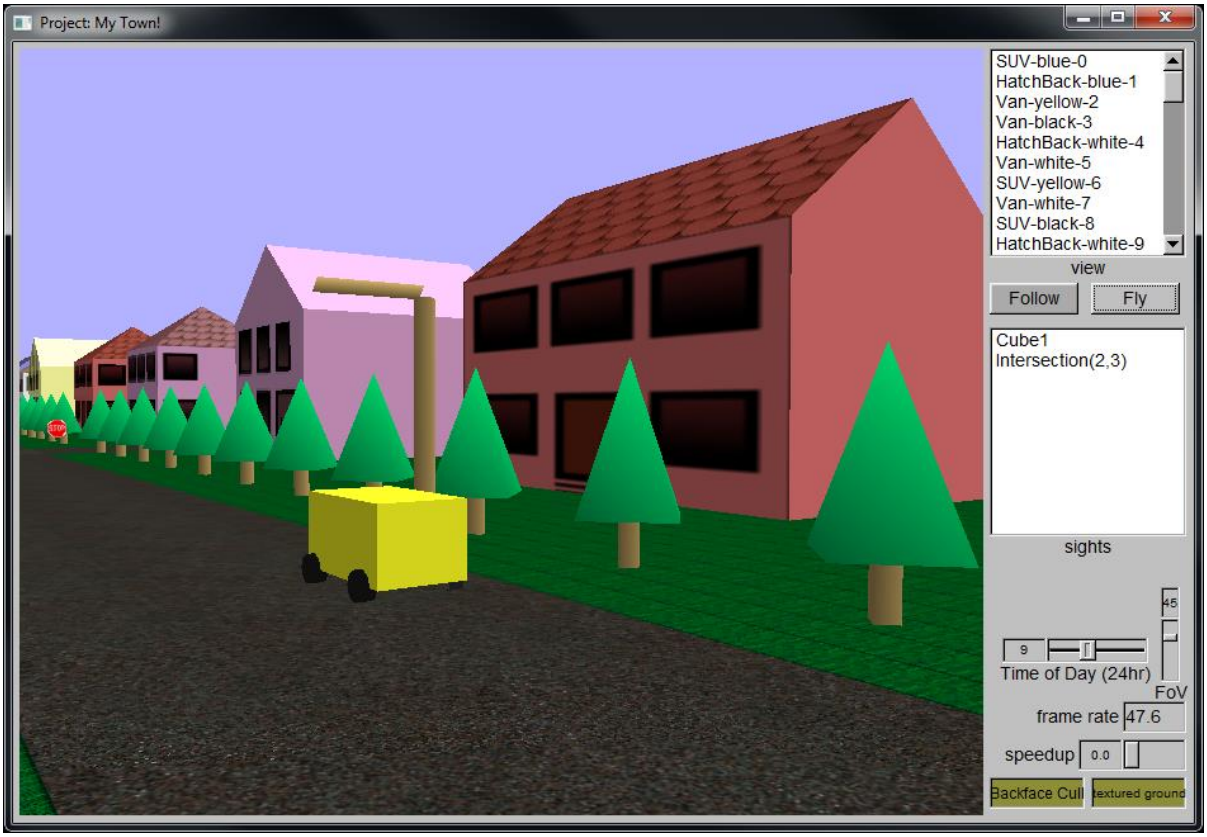
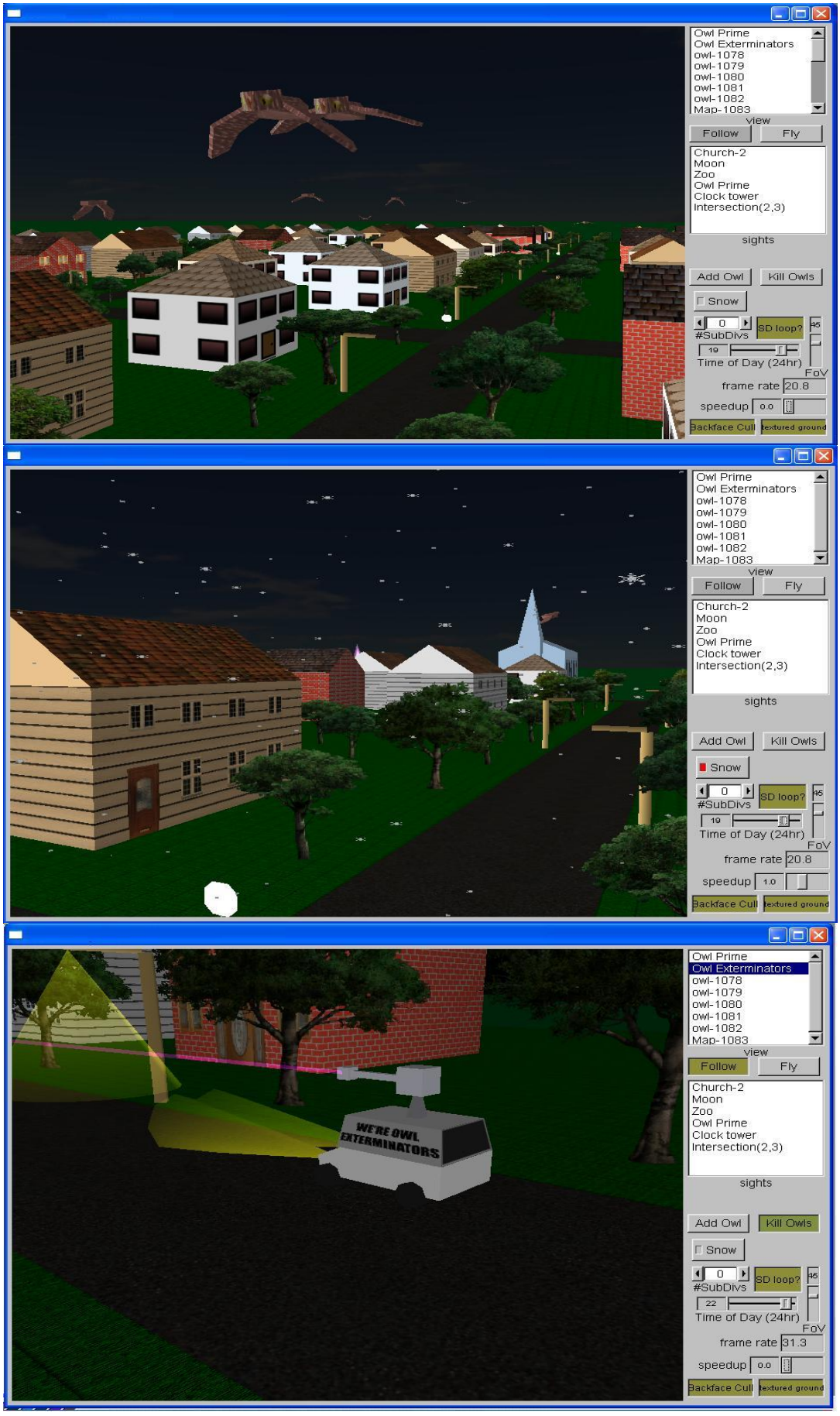


Fig. 1 The scene of the town

You are required to write your own codes to add more types of objects, textures, and movements. The program will also allow the user to explore the town from different perspectives. Snapshots of good solutions are shown below.


Owl Prime
Owl Exterminators
owl-1078
owl-1079
owl-1080
owl-1081
owl-1082
Map-1083
view
Follow
Fly
Church-2
Moon
Zoo
Owl Prime
Clock tower
Intersection(2,3)
sights
Add Owl
Kill Owls
Snow
#SubDivs
SD loop?
Time of Day (24hr)
FoV
frame rate 20.8
speedup
Backface Cull
textured ground
WE'RE OWL EXTERMINATORS

The ultimate objective of this project is to give you an opportunity to explore topics in interactive graphics: how do you make things that look interesting and interactive. Part of this project is artistically related. You need to make interesting objects with good textures for your city to look nice. Part of it is technical. You have to pick things that can be implemented efficiently with interesting behavior. We will provide a singleton framework, which includes basic control of the GUI and some basic features of the system. You can focus more on the graphics and extend the whole system with cool effects.

II. Implementation Details

Basic Requirements:

You should accomplish the following goals to get the basic points.

1. New moving objects.
   You need to add or modify multiple (at least three) types of moving objects (e.g. dogs, pedestrians, birds, helicopters, robots, etc.). OpenGL primitives and very simple objects do not count. You can start by modifying the car to different type of vehicles such as bus and taxi and adding details to make it look more realistic.
2. New behaviors for the moving objects.
   Associate moving objects with different behaviors (at least two new behaviors such as flying, jumping, hand waving etc.).
3. New buildings and sceneries.
   You have to add different types of buildings and sceneries (at least three), such as elliptical race track with two cars moving on it, fountains, museums, playgrounds, malls, skyscraper etc. The buildings and sceneries have to be distinguishable among each other.
4. New textures.
   Add textures for trees, houses, cars and your own objects. Some of the textures should not be flat (i.e. it wraps onto multiple polygons).
5. You must add something that is affected by the time of day. For example, you can have an object that changes color (the shader is sensitive to the time of day) and shape (something besides the shader that is sensitive to the time of day).

Bonus Features

You can use all the techniques learned from the course to decorate your town. Here are several suggestions. The bonus points are given depending on the excellence and difficulty of your work. Implementing challenging and diverse visual effects will get you higher score. Here are some examples:

- Incorporate animation techniques, such as particle systems, to realize smoke, fountain, firework etc.
- Use “advanced” texture mappings: multi-texturing, projective (slide projector) texturing, environment mapping, bump mapping, or shadow mapping.
- Build an object made out of a curved surface. You can implement subdivision, or some form of

parametric surfaces, or do a surface of revolution.

- Improve the lighting effects in the scene.
- Make objects follow a cardinal cubic spline.
- Traffic light system so the cars and pedestrians will follow the light signals.
- Other features are encouraged, such as anti-aliasing.

III. Framework and Files

1. Build SimCity

   We provide a segment of skeleton codes for you to start with. There are three folders under the *ProjectFramework*:

   GrTown: Includes all files (.cpp, .h, and vs2010.sln) that you need to work on.

   Utilities: Includes utility files that you may need. Do not modify them if not necessary.

   WinFltk: Includes the Fltk tool kit for the user interface. Do not modify it if not necessary.

   There is one thing to be aware of: the user interface (the widgets and whatnot) is defined in the "GraphicsTownUI.fl" file. Fluid (the FlTk UI designer) turns this file into a .cxx and .H file. Do *not* edit those files if not necessary. In case you cannot compile the solution in VS2010.net, try replacing line 9 in the GraphicsTownUI.cxx, "`static using std::vector;`" with "`using std::vector;`"

2. GUIs for SimCity

   There are sliders to control the time of day (which does not change itself), the field of view of the camera, and the speedup factor (you can set it to zero to stop time, or set it higher to speed things up).

   The "Fly" button puts the system into flying mode (the default mode). The flying interface uses the following keys:

   - Left mouse button: Steer
   - Right mouse button: Fly (If the flying gets "stuck" you will have to click the mouse again)
   - Arrows: Turn
   - Space: Forward
   - 'X': Reverse
   - Arrow keys on the keypad: Translate (Strafe)
   - NumPad5: Forward
   - NumPad0: Reverse
   - 'WASD' keys do what you might expect if you play shooter games

   The preset town is pretty boring. It is made of a small number of simple objects, and a few behaviors. Your job in this assignment is to create a better town.

3. How to start

   You can actually do most of the work without changing much of the existing code.

   1. To complete the design of objects like cars and trees, you need to modify files under "Examples Objects and Behaviors"
   2. To create an elliptical road for example, you need to modify the Roads.h / .cpp under the directory "Roads" to insert a new type of roads.
   3. To design your own objects / behaviors / shaders, you will need to modify main.cpp, and add new files for new GrObjects and Behaviors.

IV. Report

Prepare snapshots of your town in a report file **(.doc)**, including all of the added moving objects, the newly added buildings / sceneries, objects that change with time, any advanced features you have implemented, and **4** most beautiful views of your town. Please also list out who implemented the objects, behaviors, features etc.

V. Grading Scheme

Your assignment will be graded by the following marking scheme:

Basic	55%
At least two new moving objects (+3 marks each)	
At least two new behaviors for moving objects (+2 marks each)	
At least two new buildings and sceneries (+3 marks each)	
New textures (+2 marks each)	
Something affected by the time of day (+2 marks each)	
Bonus feature (+6 marks each)	30%
Report	5%
Demonstration	10%
Total:	**100%**

Note: no grade will be given if the program is incomplete or fails compilation.

VI. Project demonstration and submission guidelines

1) The project demonstration will start at **10:30am in the computing lab SHB924**. Randomized time slot for each group will be announced via eLearn Blackboard. If you are unavailable to demonstrate on that day, please upload your project to eLearn **before 10:30am** of the due date and we will arrange another day for you.

2) Please test your program in the lab machines earlier to make sure it can be executed successfully during the project demonstration.

3) A few questions will be asked during the demonstration. You will be asked to explain some of the codes in your program and discussed about how features are implemented.

4) After your demonstration, zip the whole project and your report in a .zip. Name it with your group number (e.g. group01.zip) and submit your assignment via eLearn Blackboard. (https://elearn.cuhk.edu.hk/) (Only one student of each group has to submit)

VII. Acknowledgements

The project and skeleton codes were designed by Prof. Michael, modified by XU Li.