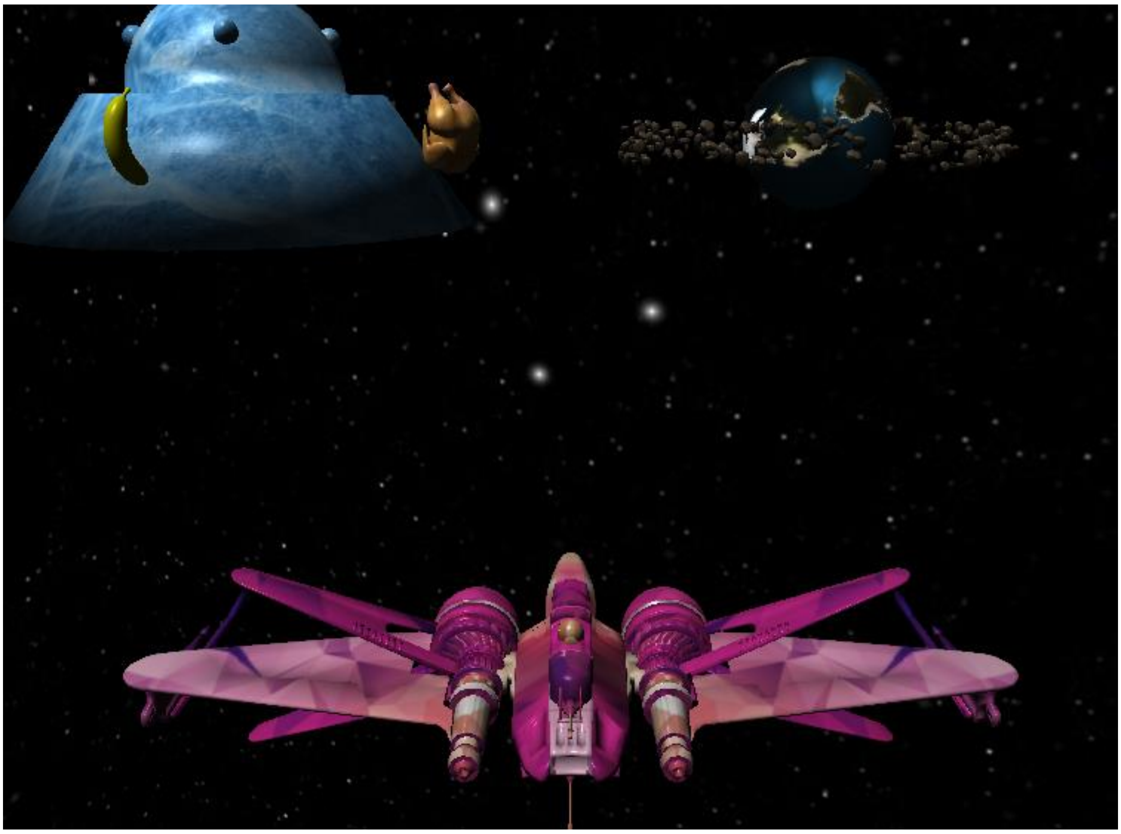
CSCI3260 Project Report

Group Members :

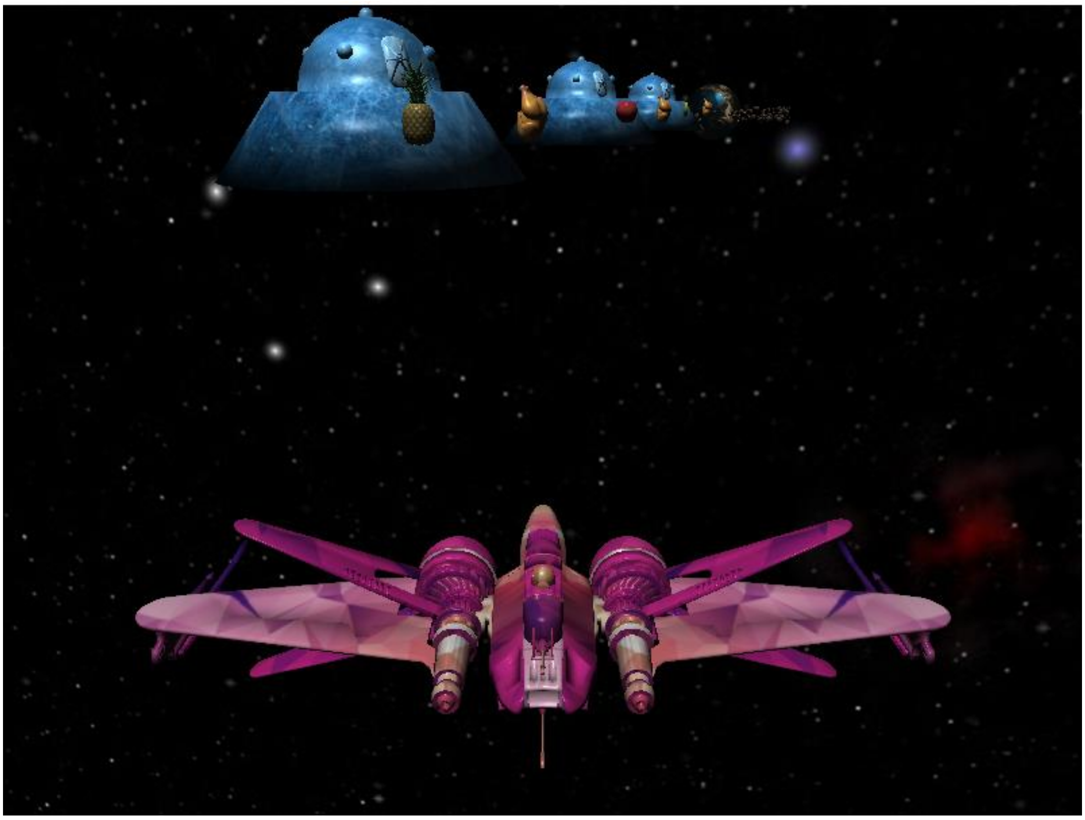
Chu Chun To 1155127149

Wong Tin Wang David 1155127053

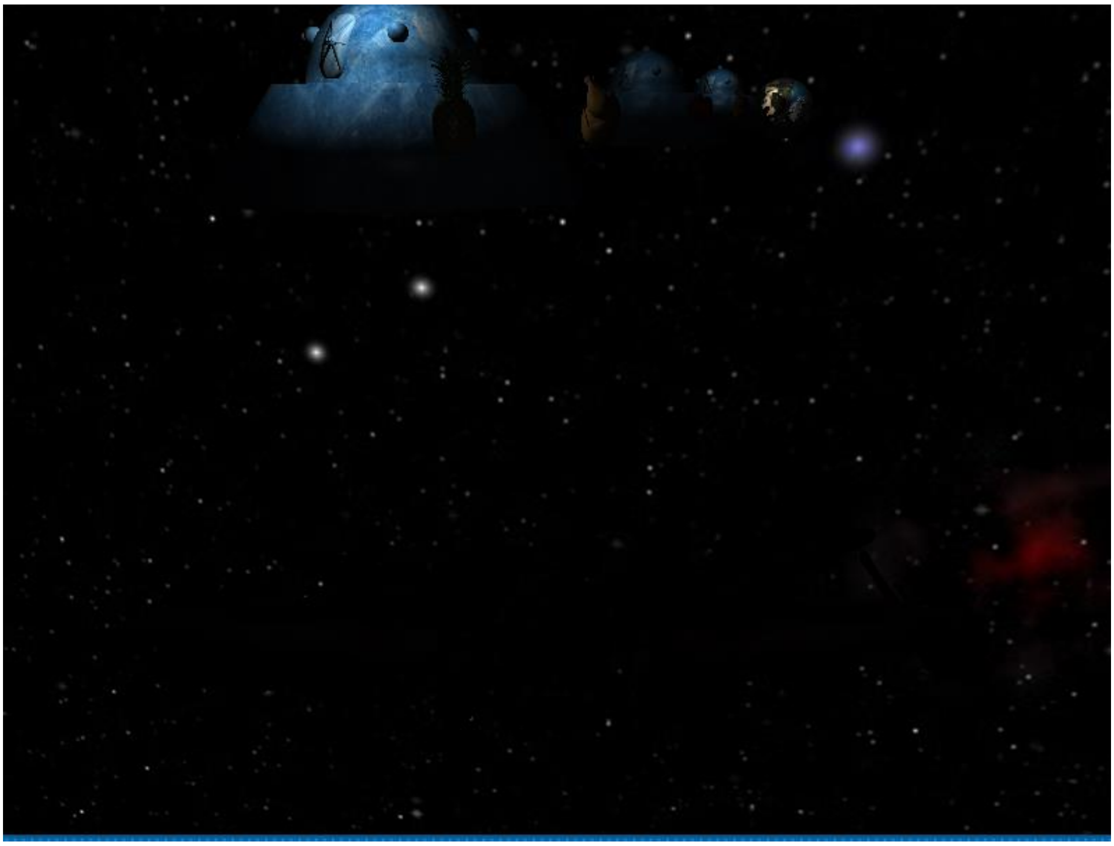
1.Overall scene:



With Both directional light and point light source:

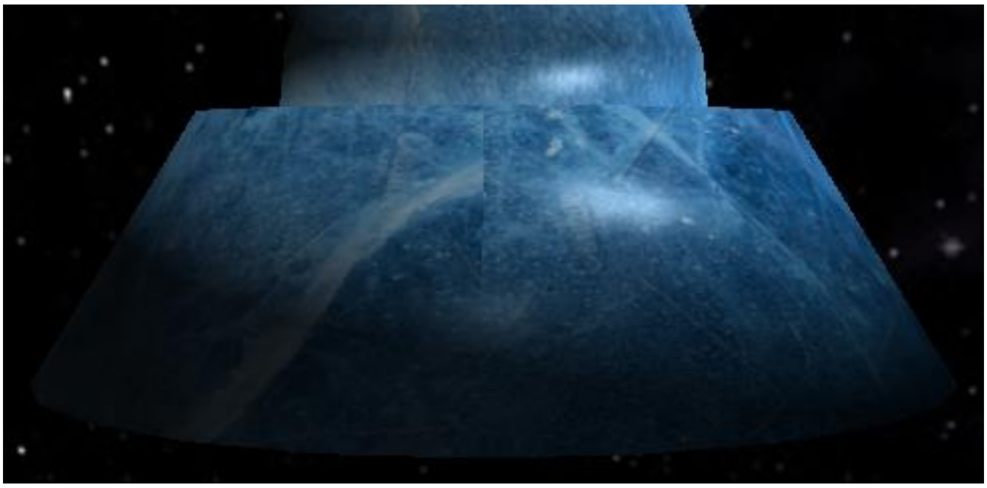


With only point light source:

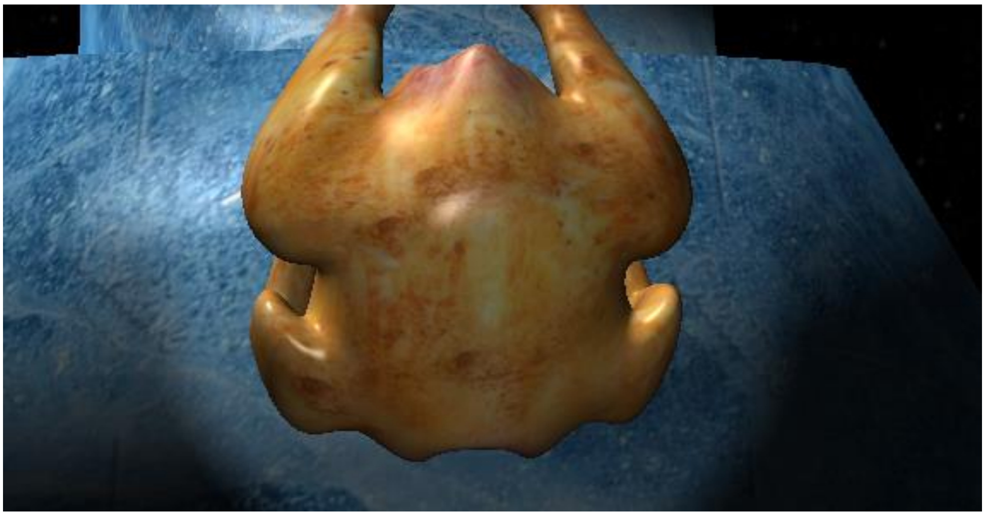


2.Basic light rendering:

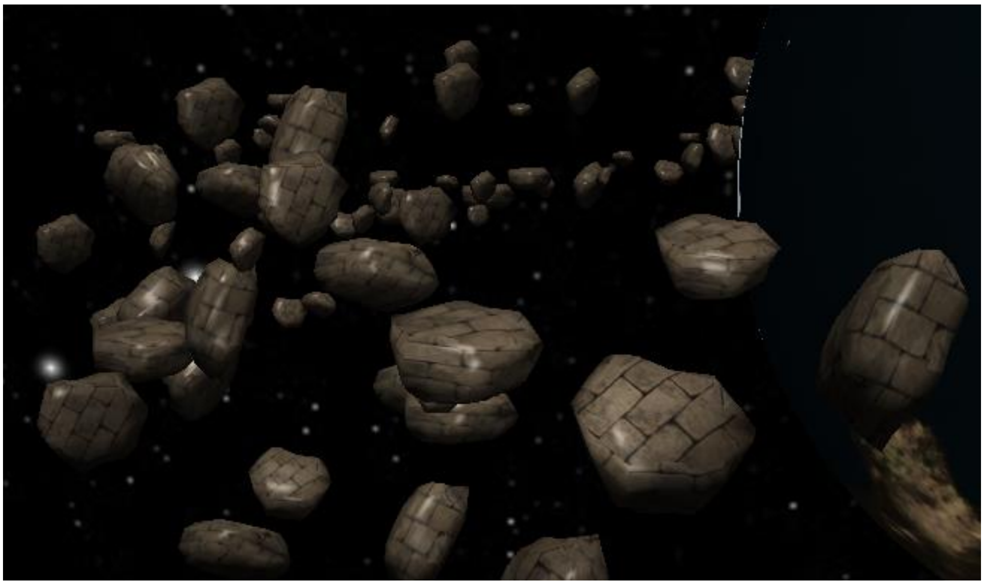
Alien vehicle:



Chicken:



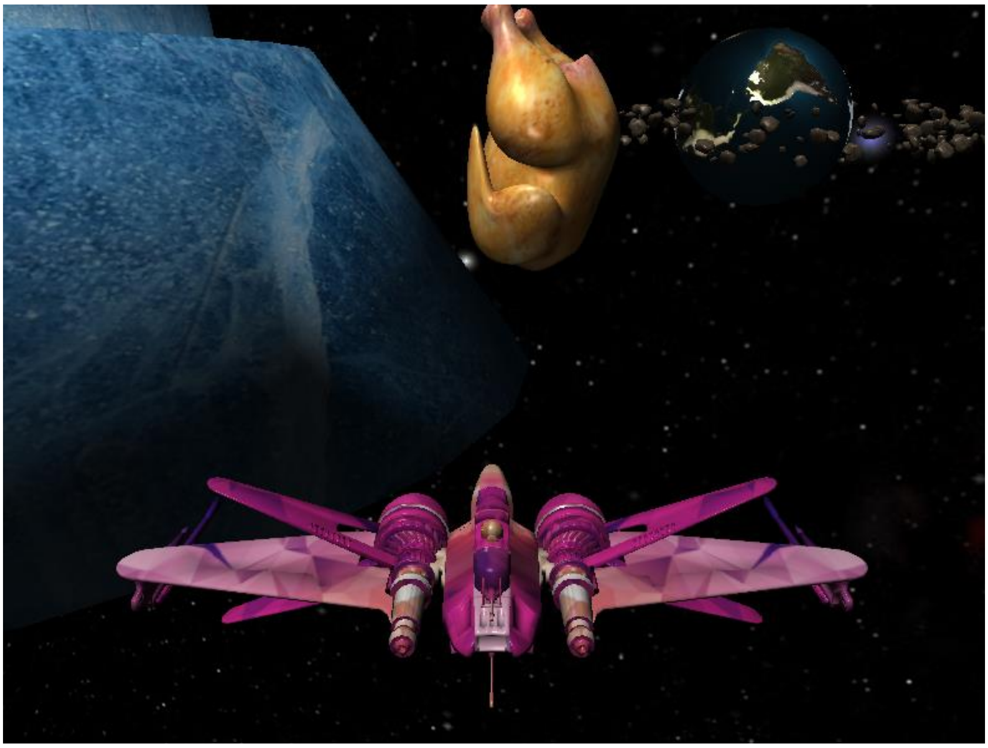
Rocks:



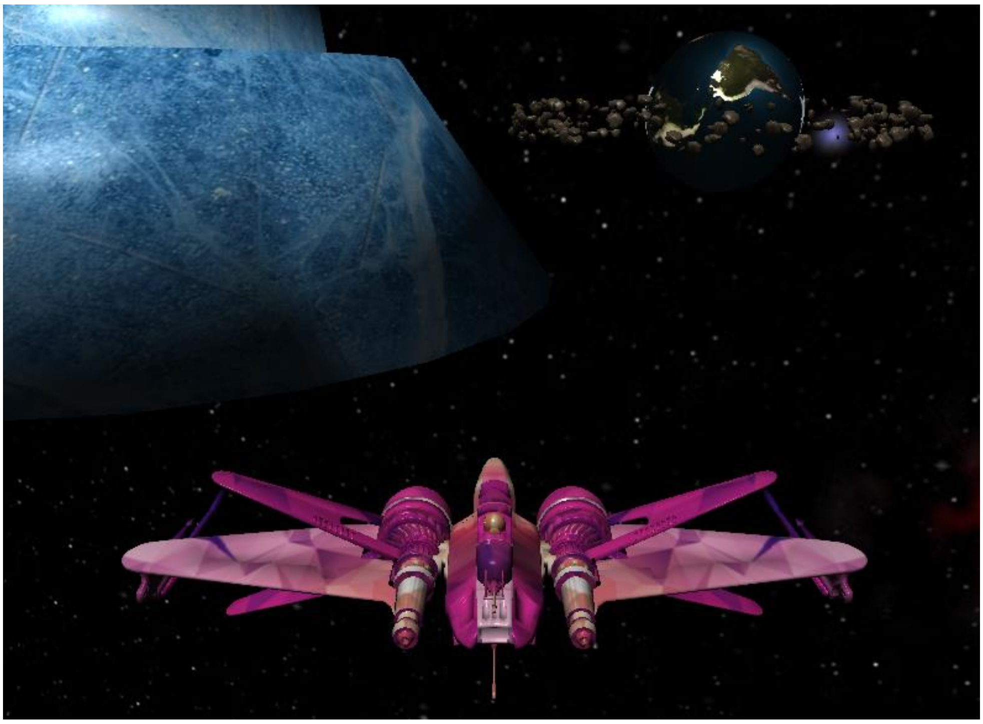
3.

Collecting foods:

Before collecting:

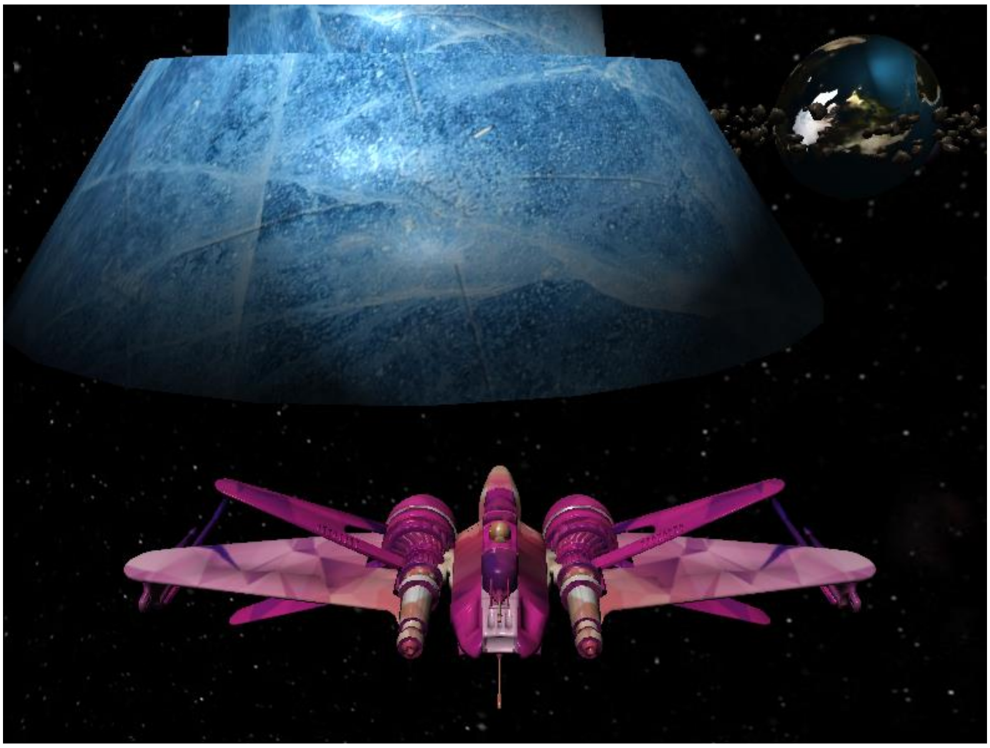


After collecting:

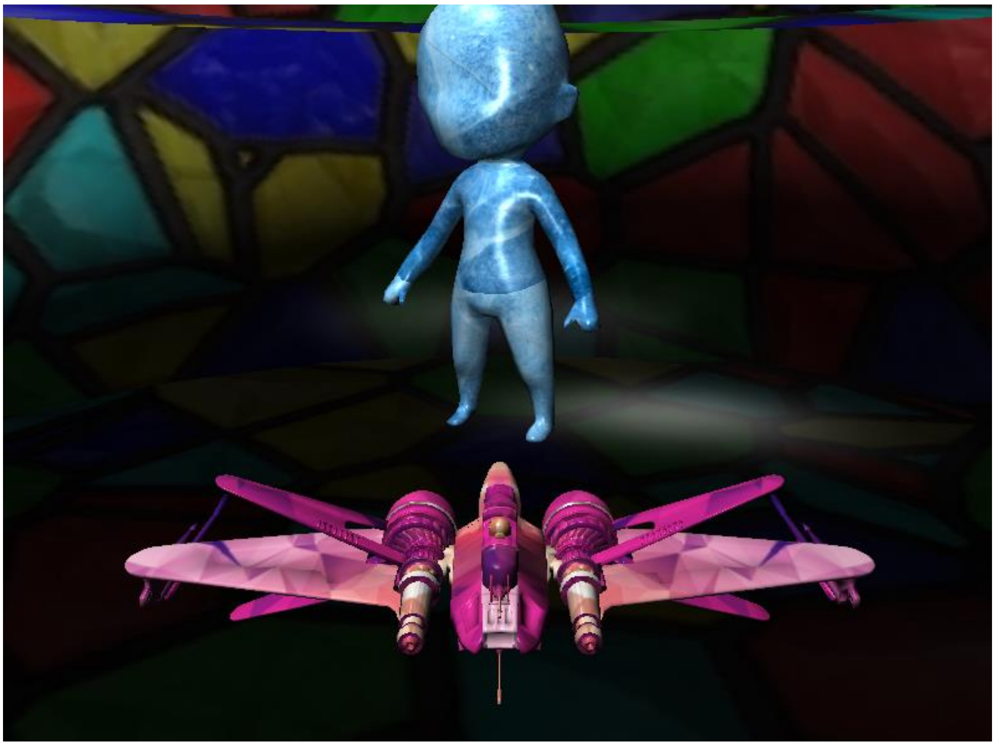


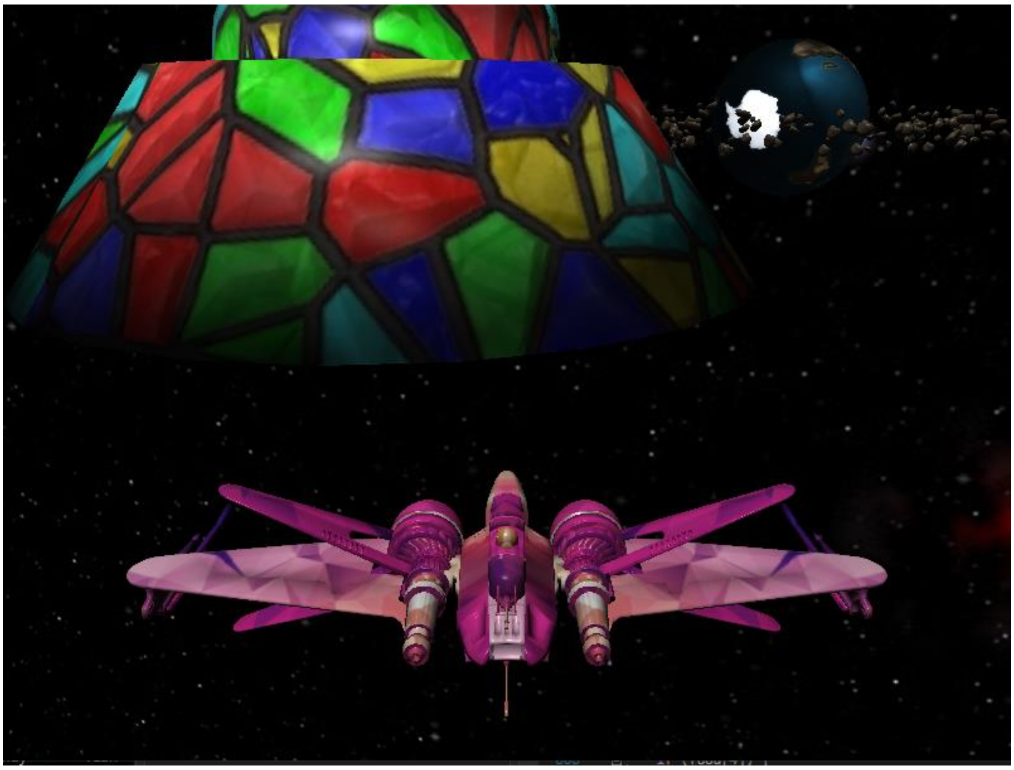
Visiting aliens:

Before visit:

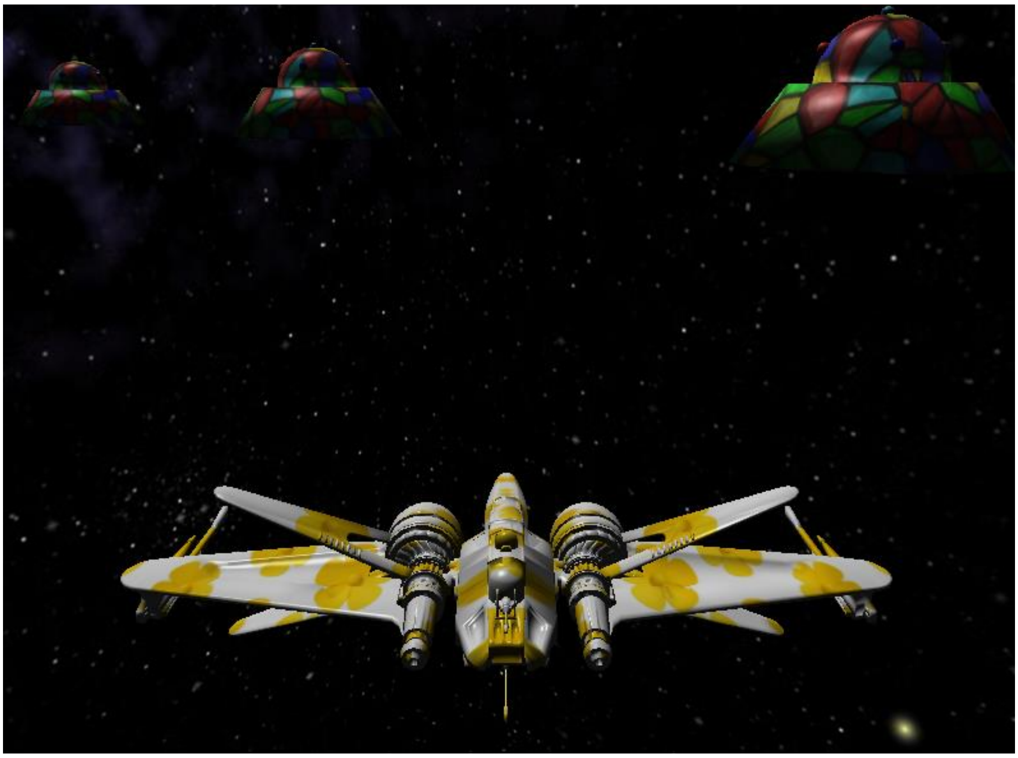


After visit:



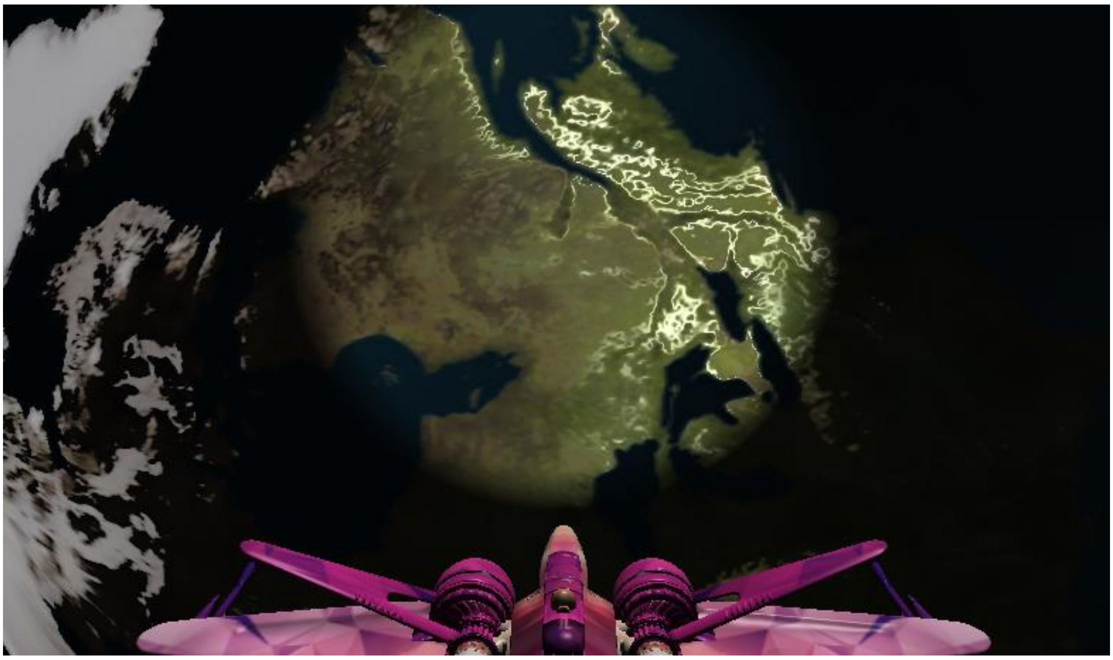


Texture change after visiting finished:

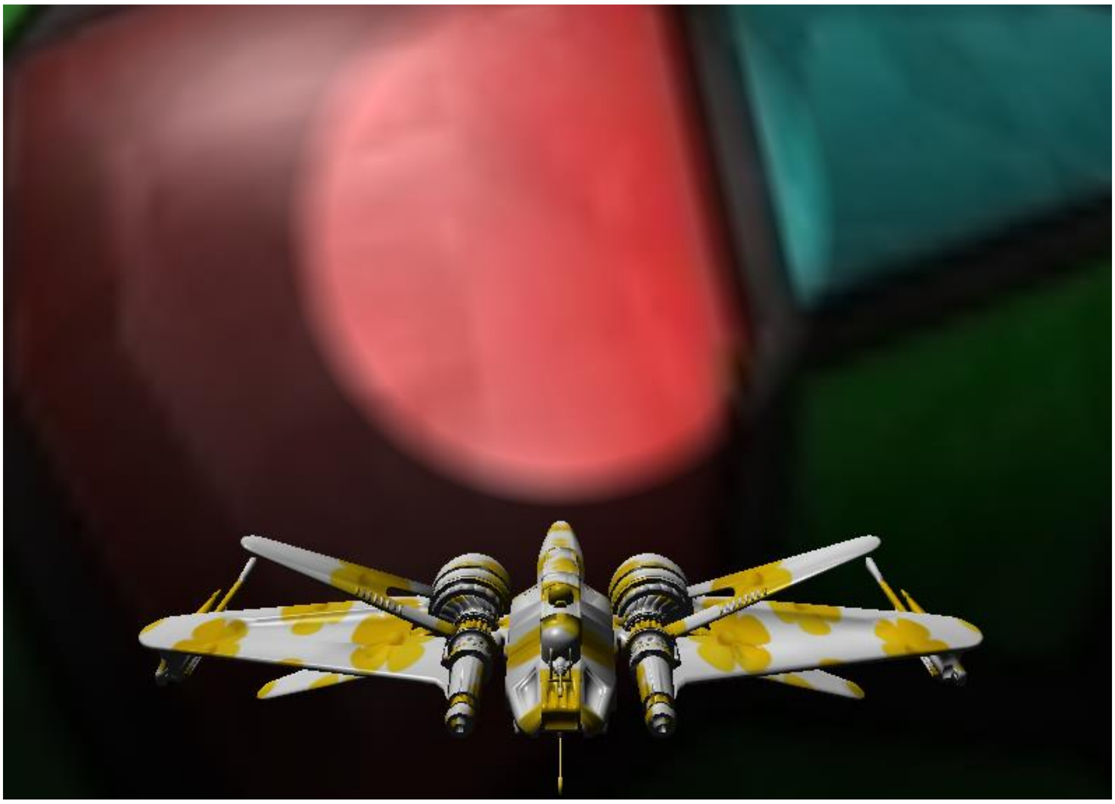


4.Bonus features:

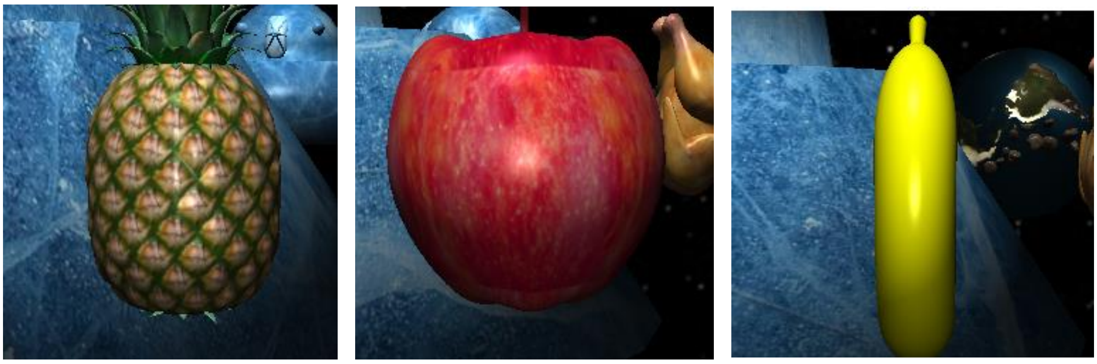
Normal Mapping:



Another light source (Spotlight):



Extra Food: Pineapple & apple & banana



5.Implementation details:

General details:

Interactions:

Mouse: Move left/right: turns spacecraft left/right

Scroll up/down: zoom in/out

Keyboard: Up/down/left/right: move spacecraft forward/backward/left/right

W/S key: increase/decrease directional light intensity

A/D key: increase/decrease point light intensity 1

Q/E key: increase/decrease point light intensity 2

Lighting:

Directional light: 1.

Point light: 4.

Spotlight: 1 (in front of spacecraft).

Extra objects/texture loaded:

Pineapple(10200_Pineapple_v1-L2.obj, 10200_Pineapple.jpg)

Apple(apple.obj, istockphoto-512401658-612x612.jpg)

Banana(banana.obj, 1200px-ICS_Quebec.svg.png)

Specific details:

Skybox(for requirement 3):

LoadCubeMap function used to load textures for 6 faces. Shader *skybox Vertex/Fragment shader* is used, which samples texture color directly without lighting calculation.

Self rotation for planet and alien vehicle(for requirement 2):

Variable selfrotating is used to keep track of rotation angle. It increments by 0.1 each time before paintGL executes and reset to 0 when reaching 360. Rotation matrix about y axis is set with this value and applied to object planet and alien vehicle.

Asteroid ring cloud(for requirement 5,6)

200 asteroids in total. Model matrices with random translation/scale/ rotation are generated. Same VAO is drawn for all asteroids but with different model matrices applied each time. They are rotating around the planet with uniform circular motion.

Collision detection(for requirement 10,11):

Calculate the vector from camera position (near spacecraft) to object world position and use magnitude of the vector as distance.

Normal Mapping (for bonus requirement 2):

Normal map of the form of rgb image is passed to fragment shader as texture and rgb value is converted to (-1, 1) interval to be used as normal for later lighting calculation.