

Project report for the CG 100433 course

Project Title

3D Balls Battle – An interactive 3D game

Team member

王维斯(Auditing)

Abstract

Briefly describe your motivation, the goal and the content of your project and its involved techniques.

Motivation

Have previously completed a 2D version in Python.

Simple programming and modeling.

Hope to participate in the project to improve my code capability.

Goal of the project

To completed a 3D version of the Ball Battle, and as much as possible to achieve some advanced features, such as network multiplayer, advanced visuals, smart bots and so on.

Scope of the project

Only using the library of windows x64, OpenGL 4.5(Nvidia).

Involved CG techniques

OpenGL v4.5(Nvidia), GLFW Toolkit, GLAD Toolkit

CG programming core techniques, project framework foundation, all implementations are inseparable from these three toolkits.

“STB Image” loader toolkit, Assimp model loader toolkit

The reading module of the picture and obj file, with these two toolkits that can quickly and easily read the picture and obj file that converts to the data type which OpenGL required.

GLM Toolkit

The full name is GLMath, which makes it easy and very efficient to perform operations between vectors and matrices.

ThreadPool

The feature of C++11, which can accelerate various complex mathematical operations and decisions to improve the gaming experience.

PBR Shader

The full name is Physically Based Rendering, which is a collection of render techniques that are more or less based on the same underlying theory which more closely matches that of the physical world.

Project contents & Implementation

Modeling the OpenGL Coordinate System ...Completed!

Initially learn OpenGL coordinate system

Render the balls/sphere ...Completed!

Use the sphere obj file to reduce the burden of generating sphere coordinates

Control the balls:

Camera control (Lng&Lat system) ...Completed!

Use the latitude and longitude coordinates to store the view position, so you can easily modify the latitude and longitude coordinates with the mouse offset to modify the camera's viewing

orientation.

Yours-ball moving control (Camera front) ...Completed

The movement of the balls can be achieved by adding the direction vector of the camera to the current coordinates of the balls.

Randomly generated the beans ...Completed!

Use the random number to generate the coordinates of the bot ball and the beans.

Kill determination system ...Completed!

Through the distance between the ball and the ball, the distance between the ball and the bean determines whether the two intersect, and if the relation of the radius is determined, the kill determination can be achieved.

Splitting module ...Completed!

Divide the ball into two and record the new ball information at the new address

Spitting module ...Completed!

Sprinkle one-hundredth of your own size into beans and feed them to other balls.

AI system (Fake AI/Pinball) ...Failed!

Due to lack of time and lack of capacity, this module is unfortunately not implemented.

The current solution is to randomly generate the motion vector of the bot ball and then hit the boundary bounce (Opposite of vector). Like pinball.

Network system ...Failed!

Due to lack of time and lack of capacity, this module is unfortunately not implemented.

Better visuals

Better lighting and materials (PBR Shader)

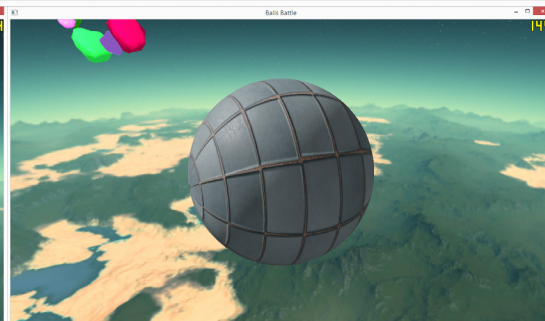
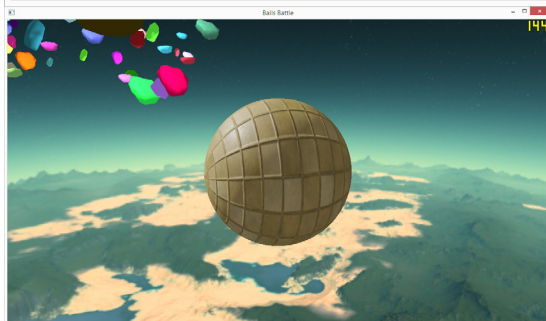
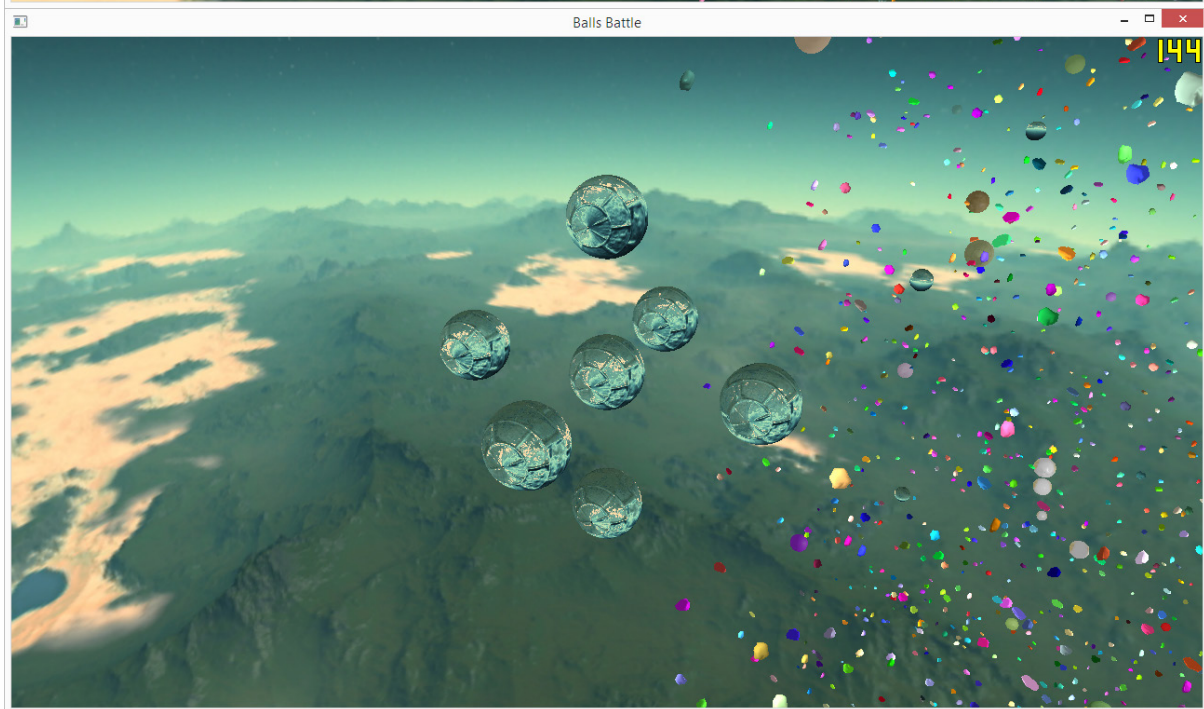
Shadow and ambient light (PBR Shader)

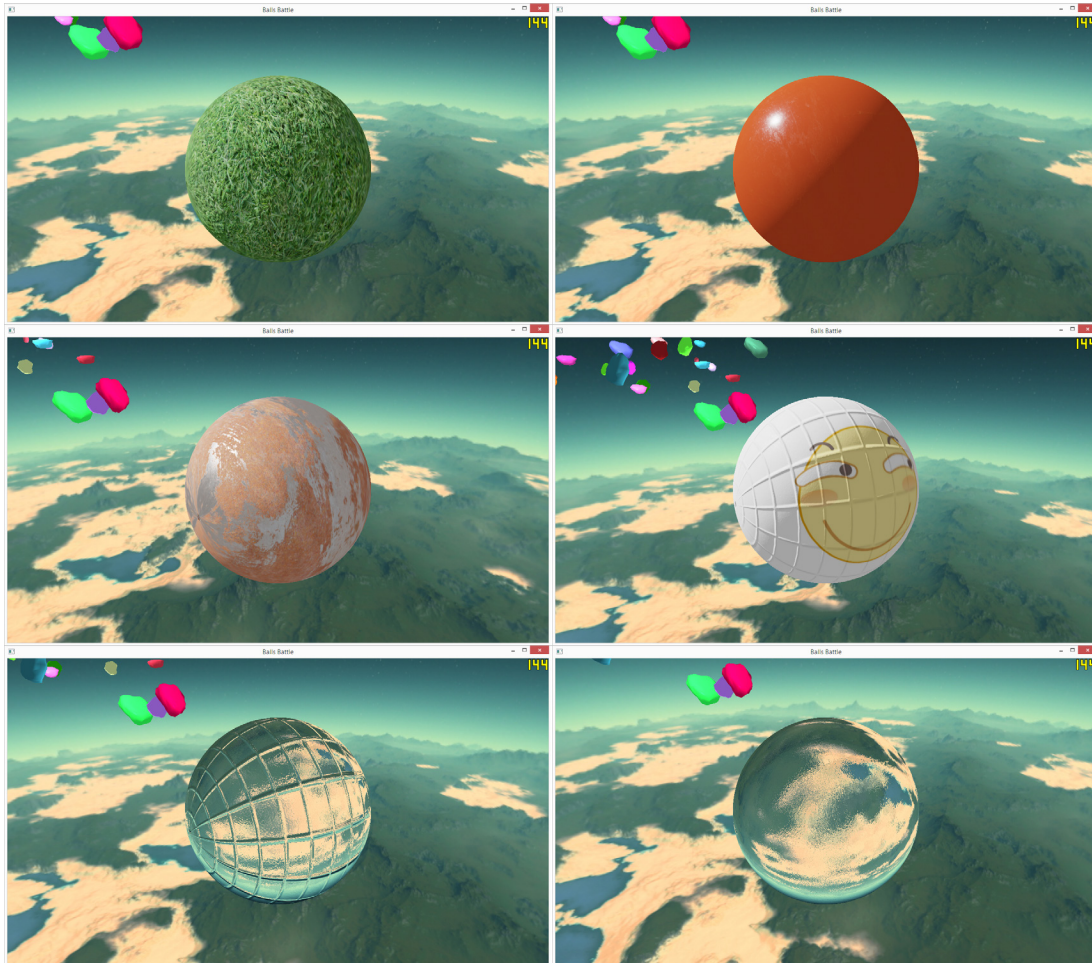
Convert to the DX12 version to Ray-tracing ...Failed!

Due to lack of time and lack of capacity, this module is unfortunately not implemented.

Results







Roles in group

王维斯 : Do all the things.

References

<https://learnopengl.com/>

Almost all of the information comes from the site, and I am very grateful to the author of this site.