

CS 680
Computer Graphics

PINBALL

Andrew Flangas
Christina Symon
Lucas Calabrese

CONTENTS

1. Overview

- i. Dependency
- ii. Extra Credit

2. User Manual

- i. Build Instruction
- ii. Keyboard and Mouse Interaction

3. Figures

4. Technical Manual

- i. Issues Faced
- ii. Things we should have done different
- iii. Changes

1. OVERVIEW

The project is programmed to create a pinball machine using computer graphics. The important methods of computer graphics used in the project are mainly model loading, texture mapping, bullet and lighting. The object for the model is built in a software called “Blender”, which is called in the program using assimp functions. The texture mapping is done using ImageMagick. Bullet is a physics engine which helps in mapping the collision of the objects. The final outcome is a basic pinball game with two paddles, collision cylinders, and a plunger to launch the ball into the game. The game also features different lighting aspects such as diffused, specular, ambient, and an option to switch to spotlight also. The game ends after three strikes and the final score is displayed at the end.

i. Dependency

Installation of three programs are required to run this project on the operating systems: GLEW, GLM, and SDL2.

Ubuntu/Linux

It works on the ECC ubuntu machine since they already have it downloaded.

```
sudo apt-get install libmagick++-dev
```

```
sudo apt-get install libassimp-dev assimp-utilsev
```

```
sudo apt-get install libglew-dev libglm-dev libsdl2-dev
```

```
sudo apt-get install libbullet-dev bullet-dev
```

Mac OSX

Installation of brew is suggested to easily install the libs. Ensure that the latest version of the Developer Tools is installed.

```
brew install glew glm sdl2
```

```
brew install assimp
```

```
brew install imagemagick
```

```
brew install bullet
```

ii. Extra Credit

- Plunger intensity changes via holding down the spacebar
- Spotlight radius and intensity can be changed with keyboard inputs.

2. USER MANUAL

i. Build Instruction

The building of the project is done using CMake, installation with apt-get or brew may be necessary. Later use with CMake and Shader files will require the copy of a directory where those files are stored (ex. shaders). To do this in the `add_custom_target` function place.

```
COMMAND ${CMAKE_COMMAND} -E copy_directory ${PROJECT_SOURCE_DIR}/shaders/  
${CMAKE_CURRENT_BINARY_DIR}/shaders
```

Cmake instructions

```
mkdir build
```

```
cd build
```

```
cmake ..
```

```
make
```

```
./Pinball
```

ii. Keyboard and Mouse Interaction

Keyboard Input

- Hold space bar- plunger launcher
 - 'Z' key - operates the left paddle
 - 'M' key - operates the right paddle
 - '8', '5', '4', '6', '1', and '2' keys on the number pad- camera controls
 - 'N' key- per fragment shader
 - 'B' key- per vertex shader
 - 'P' key- spotlight
 - 'L' key- normal light
 - 'A' and 'S' keys- ambient light adjustment
-
- '1' and '2' keys- specular light adjustment
 - '3' and '4' keys- specular light of the ball adjustment
 - '5' and '6' keys- spotlight intensity
 - '7' and '8' keys- radius of spotlight adjustment

3. *FIGURES*

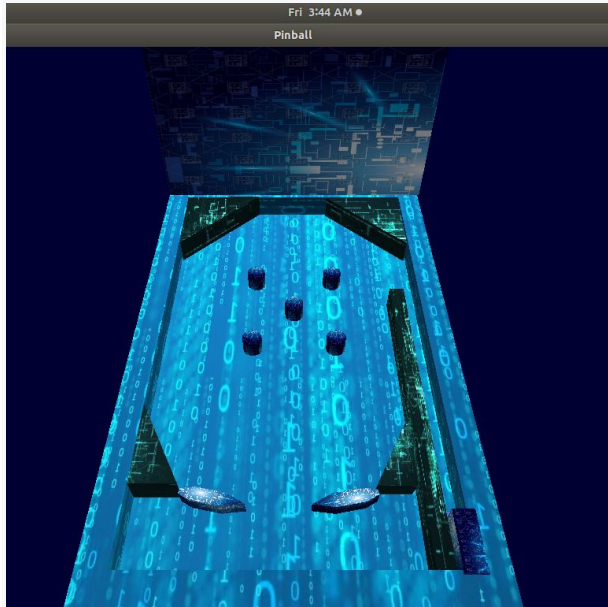


Figure 1: Default view of the pinball table

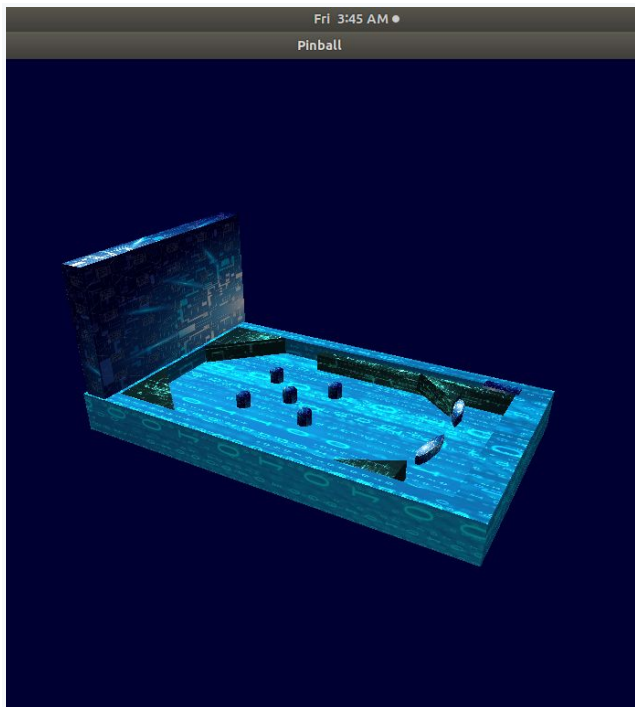


Figure 2: Side view of the pinball table

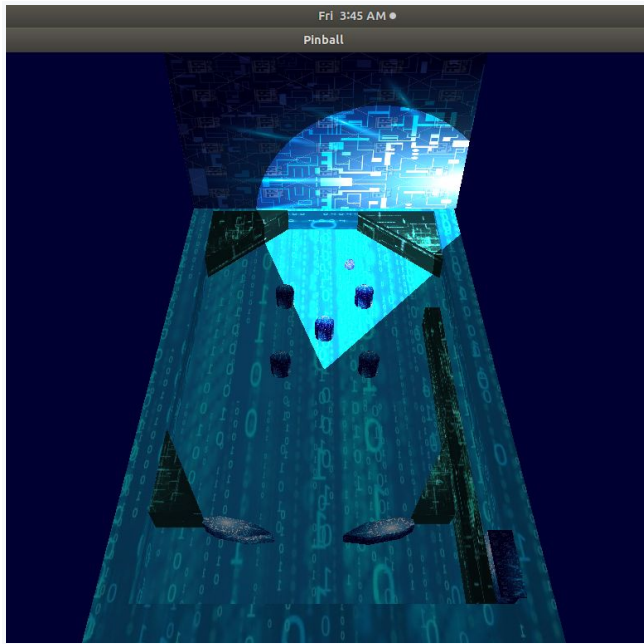


Figure 3: Image of the spotlight

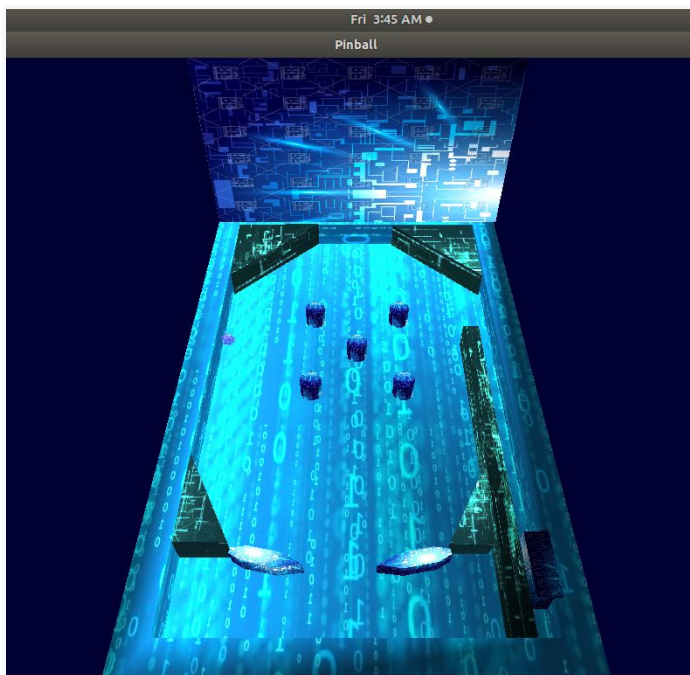


Figure 4: Different lighting scheme of the pinball table


```
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Wow! 2 points!!!  
-----  
Strike: 3  
GAME OVER  
Score: 106
```

Figure 5: Score board

4. TECHNICAL MANUAL

i. Changes Made

Added spotlights

Added flippers

Added detection between specific objects

ii. What we could have done differently

We could have used kinematic objects to do the flippers.

We could have worked in creating better pinball objects.

iii. Issues

Lighting shaders