

HW2:Direct3D First Person Shooter

(Abhishek Venkatesh: GTID#:902398759)(venky@gatech.edu)



Game Usage:

- 1) Move around the arena using the arrow keys.
- 2) Move the mouse to change view. You can even look up and down or in any direction
- 3) Jump: Press Right Mouse Button to jump. If you do this while you are moving you can jump over the monsters to avoid them.
- 4) Fire: Use left mouse button to fire bullets. You can use multiple bullets. You can aim using the cross hair in the middle of the screen
- 5) The Gun always points to the cross hair(sphere in the middle of the screen). It adjusts to the mouse movements and is made view independent.

If a bullet hits any of the monster, it dies and re-spawns from the ground from a near about random location. The monsters always move towards the player and try to kill the player. If any of the monster comes close to you, you will die and the game starts again (You will go back to starting position).

Extra Credit features:

- 1) The game includes sound (DirectSound)
- 2) Multiple Bullets and multiple Monsters
- 3) Gun is shown(it is made view independent)
- 4) Cross hair is shown
- 5) Player can jump over the monster (right click).
- 6) Camera view is controlled using mouse movements(Like popular first person shooter games)
- 7) The Gun and the arena are actually X-files. So you can create your own arena and load it in the game (Write now this has to be done in the code so will require you to re-compile).

Folder Description

- 1) The complete source code and VC++ project is given under “CompleteSourceCode”
- 2) The executables and resources are given under “GameBinaries-Executables”. You can run the game by double-clicking FPShoot.exe

Compiling and Running Instruction

- 1) The VC++ project is included in the source code provided (see folder CompleteSourceCode). You will probably have to set the path for including the DirectX directories and libraries. Otherwise there are no special instructions to compile the code
- 2) The executables are given in the folder “GameBinaries-Executables”. To run the game simply double click “FPShoot.exe”

Game Design

The game uses the following objects to model the game. In the source code each object is represented by a separate class

- 1) Camera(Position, View, ViewUp, Fov, front face, back face)
- 2) Map(or arena)
- 3) Monsters
- 4) Gun (Player is assumed to be a gun).
- 5) Bullets

Each of the above objects (except camera) is derived from a base class object CObject. Class CObject provides the functionality of loading an X file, setting the transforms and rendering the object. Any of the derived classes can override the rendering function to suite its needs. Each object defines what happens when it collides with something. So in short the loading and rendering of an Object is encapsulated in class CObject and the logic of the Object can be customized based on what that Object represents. Each Object also stores its current position and orientation (looking at vector).

Besides these the game uses four light sources, a small sphere in the middle of the screen for a cross hair used for aiming. The player may choose to avoid the monsters by jumping over them instead of shooting them using right click.

Collision Detection: DirectX provides support for generating bounding boxes for each mesh. To Speed up the collision detection we simply check if any 2 objects bounding boxes intersect (it is very cheap).

Bullet movement: At the instant when a bullet is fired, we store the direction in which it was fired and the starting position of the bullet. In each subsequent frame we update the position of the bullet by

moving it across the stored direction (starting from the initial position). In each frame we check if any of the bullets has hit some object. If yes then we make that bullet inactive and it will not be rendered. The Game supports multiple and unlimited bullets.

Sound is added in the game for the following events

- 1) Monster trying to scare you
- 2) Bullets hitting the monster
- 3) Gun Fires

References:

[1] <http://www.robthebloke.org/> This has some demo code for playing .wav files using DirectX

[2] <http://users.ece.gatech.edu/~lanterma/mpg/>

[3] http://www.toymaker.info/Games/html/load_x_simple.html

[4] <http://www.turbosquid.com/>