T2-21-22-CS 606 / Computer Graphics May 10, 2022

Project Report

Instructors: Prof. Jaya Sreevalsan Nair, Prof. T K Srikanth

1 What kind of distortions do you notice with the texture maps you have used? What would be your approach to correcting them?

The texture was mapped onto the football field. There were no distortions since the texture was rectangle and the football field was also rectangle. When we zoom in the camera towards the football field, we could see that it is an image projected onto the surface and the surface was not irregular.

We can correct this by remapping the texture to the zoomed area i.e using Bit-mapped textures. To create a feel of irregularities on the surface of the field, we could also use Bump mapping technique.

2 Provide a brief writeup (1-2 pages) on the design you used for:

2.1 Scene graph organization

The whole game has been organised as objects in a scene, where some objects have certain types of relationship with other objects. In our implementation there is a connection between the first player camera, spotlight on a player and the player, some other objects that have a spot light fixed on them are fixed, rest of the objects are are also fixed. The hierarchy between the player, its spotlight and the camera is that as the player moves the camera follows it facing the direction of motion and also the spotlight follows the player. However the same approach is not used when the third person camera is used, it is disjoint with the player movement, but the player movement direction is constrained by the direction in which the third person camera is facing.

2.2 How the position and orientation of lights and the avatar's camera are calculated

Initially when game starts the players are given a fixed position. The light was set to the initial position of the player. The player does not move in the vertical axis, hence its y-axis position is fixed. The first person view camera is also fixed to the player's head and is set to look in the direction where the player is viewing. As the player moves the camera position is also updated, along with the position of the spotlight, it looks down on the player from a fixed height. For the third person camera which is fixed to look towards the origin irrespective of its position, it can be moved using mouse controls but it will always point towards the origin. In this mode of camera the player's direction of motion is calculated with respect to where the camera is pointing towards. The motion of spotlights remains the same as it was in the first person camera mode.

2.3 Computation of animation including collision detection/avoidance

For collision, we created a bounding box for every object in the scene (except the lamps). This bounding box is a cylinder. We calculate the distance between the two cylinders of the objects to detect collision. This collision detection function is kept in the animate loop to keep track of collisions at every frame of the game. Using this collision detection, we can do operations such as blocking the player and bouncing the ball off the surface. For bouncing the ball off the surface of the obstacle, we calculate the direction of the ball and give the appropriate directions after the detection of collision.

To animations were loaded from mixamo website. The model were in the form of fbx files which also contained the animation data for different movements. The movements for different motions were captured in the actions class object in Three.js, these were integrated in the animation class for each player. For transitions between animations a variable ”fadeDuration” has been used, it defines the steps in which the present animation will come to a stop and the next animation will begin based on the input provided. Same number of keyframes as present in the fbx file were used. Collision between player and the other objects have also been detected using cylindrical bounding boxes around the player.