Metrix Hell

Group Project Report

COMP3501

Weihang Chen

Heqing Wang

Kaiwen Mao

Brief Idea

Our game start with an idea that is eagle catch chicken, then we add hen and drone to protect chickens, and we give different types of attack to eagle. In order to make the game more playable, we add resources, which is hp and energy, that player need to care about during the game. The game take place in a farm, when the game begin, player acts as an eagle flying around the game world, the eagle needs to keep catching chicken, otherwise it will die by hunger, hens will protect their children when the eagle try to catch them, and farmer sends some drones to stop the eagle. The eagle need to dodge hens and drones, at the same time catch chickens.

Game Manual

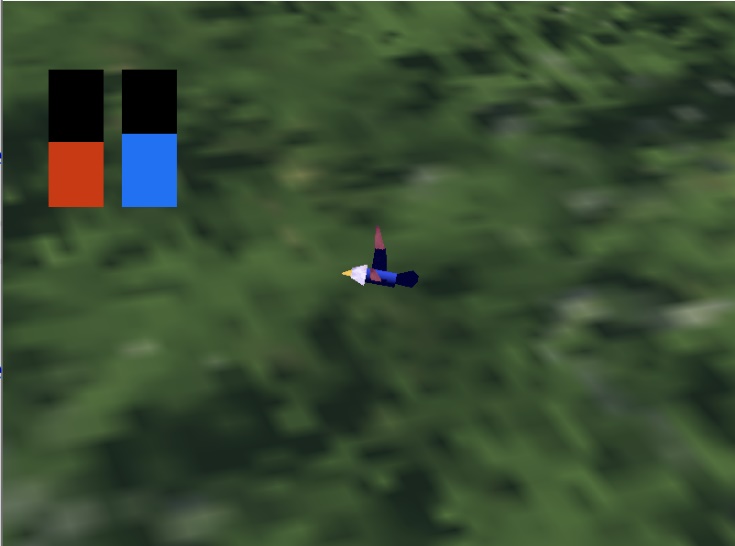
Our program starts with a start scene, press enter to start the game. Arrow keys can be used to move the sight around. ‘A’ key is used to moving forward and ‘Z’ key can slow down the speed. ‘S’ and ‘X’ keys will rotate the eagle. Player can press “tab” key to switching between first-person and third-person camera. ‘V’ key will cause a melee attack, ‘F’ key will shoot a feather to the forward, and ‘T’ key will change to a vertical down view, during this time, player can use ‘I’, ’J’, ’K’, and ‘L’ to choose a position, and press ‘T’ again to confirm. After player confirmed, a tornado will be create at that position. ‘Q’ key will quit the program immediately.

There are two important resources for the eagle, hp and energy. There are a hp and an energy bar on the screen, hp and energy bar will start with half. Hp will decrease and energy will increase through time. When hp go down to 0, the eagle will die by hunger and the game will end. If the eagle successfully caught a chicken, it will restore some hp, so the eagle needs to keep catch chicken to make itself alive. Except the melee attack, using the other two attacks need energy, feather attack needs less energy and Tornado needs more. How to manage these two resources will be task for player.

In order to catch a chicken, the eagle need to come close to the chicken and case a melee attack, when there are chicken inside the attack range, the chicken will be caught and disappear, but if there is any hen inside the attack range, the eagle will not able to catch the chicken, in addition, it will be stunned for a few seconds. Collision with a drone will also cause stun, using feather attack can destroyed a drone, when drone get destroyed, after a certain time, there will be a new drone coming out to chase the eagle. The total number of drones will increase through time. When the number of chickens is lower than a certain number, there will be a group of hens and chickens coming out from a house. Since hens cannot be destroyed, the game will become harder with more hens on the ground.

Technical Requirement

The Eagle:

The eagle is the most important object in the game. Player can use keyboard to control the eagle’s movement. Player can change camera between first person and third person, this will not affect other parts of the game, so player can just choose the way they like better. Player can control the eagle flying inside the game world, but there is a limitation on max and min height. The transformation of the eagle is hierarchy, the body part is set as root node, the head , the inside part of the wing and tails are set under body, mouse is set under head and outside part of the wing is set under inside part of the wing.

The eagle has three types of attack, Peck, the melee attack, Shoot Feather, the range attack, and create a Tornado, the special attack. The melee attack has a very small effect area, and it is the only way to catch a chicken, so player should control the eagle come close to a chicken to catch it. Feather attack will shoot two feathers, the feather will stun a hen or destroy a drone if it hit them. Tornado will ask player to choose a attack position, and create a tornado at that position after player confirm. Tornado will drag all the chicken to the middle and stun all the hens. We also add particle system to these two attacks to increase visual appearance.

*Feather attack*

*Choose position*

*Tornado*

Hen and chicken:

 Hen and chicken are ground units, which means they only move on the ground. Hen and chicken both have some basic AI behavior. Hen and chicken will roam around the space, when the eagle fly close to a certain distance, hen will move to the eagle’s position to protect chickens, and chickens will try to run away from that position. Hen will be stunned when hit by a feather.

Drone:

 Drones are flying units, they will try to stop the eagle from catching chicken. Drone also has some basic AI behavior, it will find the eagle’s moving position, and move forward to that position, when it reach that position, it will find the new position and move toward it again. If a drone hit the eagle, the eagle will be stunned. Drone will be destroyed when hit by a feather.

*The vision will be blurred, and player cannot control the eagle when get stunned*

Game World:

草地上有水

描述已自动生成 The game world is a skybox, there is a ground object inside the skybox, there are also some puddles on the ground, puddles have a reflection surface. There is a house on the ground, hens and chickens will come out from the house.

Technical Notes:

We use a main function to control all the game running. Inside the main function, the first part is initialization, it first initialize the opengl resources, then it call resources\_manager class to initialize all the resources, after that it initialize all the variable that is needed during the game, last it initialize all the scene\_nodes and add them to the scene\_map. There is three types of scene\_nodes in our code, common, skybox and camera. All the moving object in the game, like eagle, are count as common nodes, skybox node and camera node only stand for skybox object and camera object . When the initialize finished, it comes to the main loop of the game,

Inside the main loop, we first get all the user’s input, then call scene\_map’s update function, this function will call the update function for every scene\_nodes it stored. Objects that extend from scene\_node have their own update function, so that them can have different movement, this part include collision detect and destroyed object removement. After all the objects are updated, the program will call the render function of scene\_map, scene\_map will call render function for all scnen\_nodes to update their behavior on the screen. Different object also have different render method which bind with different shader to show different vision effect.