

一、Member contribution (at most two in a group):

4108056007 資工三 鄧雅文 (100%)

負責貼圖、人物繪製、星星道具製作

4108056029 資工三 王傳馨 (100%)

負責音樂、小女孩繪製、場景、盾牌道具製作

二、Goal

寫出一款將魷魚遊戲這部影集中的遊戲場景、內容重現的 3D 遊戲。

三、Introduction

我們用 Minecraft 的風格去重現魷魚遊戲中的遊戲角色，並且在 123 木頭人這個遊戲基礎之上增加道具以增加遊戲難度。

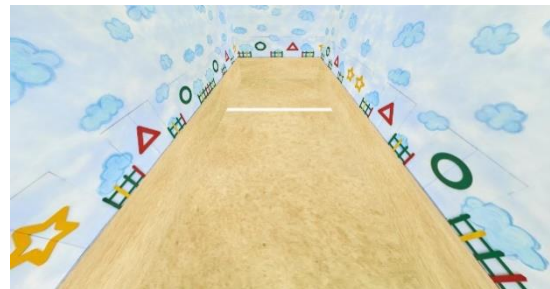
四、Code description

[場景]

- 四面貼圖，各貼兩張，用原圖和鏡像輪流貼，來消除兩個相鄰圖片間的不連續感。



(a) 平視



(b) 俯視

[選單頁面]

- 第一種選單：整個畫面背景都是一張貼圖。起始畫面的藍色按鈕框、教學頁面的道具圖案是另外繪製的。



(a) 起始畫面



(b) 教學頁面

- 第二種選單：畫面背景是用 Blending（使用作業六的 Blending），畫一個離相機很近的平面大圓，將大圓做 Blending。



(a) 死亡畫面



(b) 勝利畫面

```
glPushMatrix();
glDisable(GL_DEPTH_TEST);
glEnable(GL_BLEND);
glBlendFuncSeparate(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA, GL_ONE, GL_ONE_MINUS_SRC_ALPHA);

if (pass == 1) glColor4f(0.0f, 0.53725f, 0.63922f, 0.5); //成功通關
else glColor4f(0.54509f, 0.0f, 0.0f, 0.5); //死掉
glTranslatef(meX, meY, meZ);
glTranslatef(0.0, 0.9 - meY, -10); //初始位置

drawFilledCircle(0.0, 0.0, 80.0); //blend整個畫面

glDisable(GL_BLEND);
glPopMatrix();
```

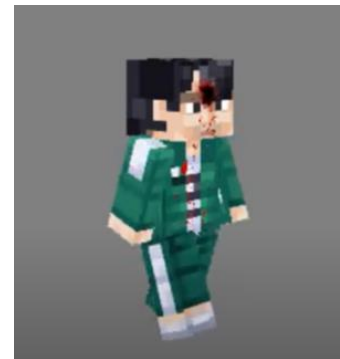
(c) Blend 畫面 code

[人物繪製]

- 採用多個長方體來繪製人物。當初沒有使用 `glutSolidCube()` 是因為我想在長方體每面貼不一樣的貼圖，所以一面一面畫，再把他們拼成長方體。
- 貼圖圖片來源擷取自網路上別人做的 **Minecraft skin**。其中正面的部分我把人臉再加上槍傷，因為（沒死掉的情況下）遊戲視角只看得到後背，只有在死掉後才會露出正面，所以就先把重槍的傷口貼上去。

```
/* 畫頭 */
static void drawHead(GLfloat size, GLenum type)
{
    GLfloat v[8][3];
    GLint i;
    v[0][0] = v[1][0] = v[2][0] = v[3][0] = -size / 2;
    v[4][0] = v[5][0] = v[6][0] = v[7][0] = size / 2;
    v[0][1] = v[1][1] = v[4][1] = v[5][1] = -size / 2;
    v[2][1] = v[3][1] = v[6][1] = v[7][1] = size / 2;
    v[0][2] = v[3][2] = v[4][2] = v[7][2] = -size / 2;
    v[1][2] = v[2][2] = v[5][2] = v[6][2] = size / 2;

    for (i = 5; i >= 0; i--) {
        switch (i) {
            case 0:
                glBindTexture(GL_TEXTURE_2D, texture[2]); //right face (model的視角)
                break;
            case 2:
                glBindTexture(GL_TEXTURE_2D, texture[3]); //left face
                break;
            case 4:
                glBindTexture(GL_TEXTURE_2D, texture[0]); //front face
                break;
            default:
                glBindTexture(GL_TEXTURE_2D, texture[1]); //hair
                break;
        }
        glBegin(type);
        glNormal3fv(&n[i][0]);
        glTexCoord2d(0.0, 0.0); glVertex3fv(&v[faces[i][0]][0]);
        glTexCoord2d(0.0, 1.0); glVertex3fv(&v[faces[i][1]][0]);
        glTexCoord2d(1.0, 1.0); glVertex3fv(&v[faces[i][2]][0]);
        glTexCoord2d(1.0, 0.0); glVertex3fv(&v[faces[i][3]][0]);
        glEnd();
    }
}
```



(b) 人物 model 正面視角

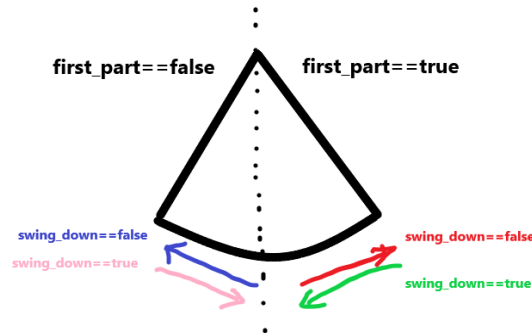
(a) 畫圖(b)的頭的例子

[人物走路動畫]

- 左手和右腳、右手和左腳，這兩個組合的擺動角度一樣。
透過改變角度、call `glRotatef()`，來旋轉人物的各部位。
- 整個擺動分兩半，每半又分成 `swing up/swing down`：
 - 如果回到原點，就改變 `first_part` 的狀態
 - `swing down` 和 `swing up` 的旋轉方向相反

```
//手腳擺動
if (swing_down) {
    swing = (swing - 10) % 40;
    if (swing == 0) {
        if (first_part == true) first_part = false;
        else first_part = true;

        swing_down = false;
    }
}
else {
    swing = (swing + 10) % 40;
    if (swing == 30) {
        swing_down = true;
    }
}
```



(a) code inside `glutIdleFunc()`

(b) 鐘擺示意圖

[小女孩轉動]

- `int girl_face_angle`：小女孩的轉動角度
- `bool kill`：現在移動是否會被殺
- `int kill_time`：控制轉到正面和背面時要等待一下再繼續轉

```
case MUSIC_ROTATE:
    //小女孩轉動
    if (!start_menu && button_select==0) {
        if (kill_time <= GIRL_WAIT_TIME) { //在正面和背面都要等待20*100毫秒(GIRL_WAIT_TIME)
            kill_time++;
        }
        else {
            girl_face_angle = (girl_face_angle + 30); //每次轉動30度
            if (kill == 1) { //如果剛剛的狀態是殺(正面)
                turn_to_kill = false; //現在就是要轉向背面
                kill = 0; //設為非殺模式
            }
            if (girl_face_angle >= 359 && turn_to_kill) { //現在正在轉到正面(殺模式)，且轉到正面了
                kill = 1;
                kill_time = 0; //在正面要重新計時20*100毫秒
            }
            else if (girl_face_angle >= 170 && !turn_to_kill) { //現在正在轉到背面(非殺模式)，且轉到背面了
                kill_time = 0; //在背面要重新計時20*100毫秒
                engine_game->play2D("audio/girl_song.mp3", false); //開始播放小女孩唱歌音樂
                turn_to_kill = true; //下次要轉向正面(殺模式)了
            }
            girl_face_angle = (girl_face_angle) % 360;
        }
    }
    glutPostRedisplay();
    glutTimerFunc(100, timerFunc, MUSIC_ROTATE); //100毫秒刷新一次D
    break;
}
```

[小女孩殺人判定]

- 如果沒有被盾牌保護時移動 (呼叫 move()) 就會死 (die = true)

```
void move(int m) {  
    if (kill == 1 && die == false && !(unbeat || shield1 || shield2 || shield3) && !pass) {  
        die = true;  
    }  
}
```

- 判定死亡後，先開槍，再倒地，再出現死亡頁面

```
//人物死掉  
if (die && shoot == 0) {  
    shoot = 1;  
}  
//開槍  
if (shoot==1 && guide==false && start_menu==false) {  
    engine_gun->play2D("audio/shoot2.mp3", false);  
    shoot = 2;  
}  
//向後倒地  
else if (shoot == 2) {  
    if (die_angle > -90) die_angle = die_angle - 10;  
    else {  
        die_menu = true;  
        walk = false;  
    }  
}
```

[道具]

- 數量：三個盾牌、三個星星
- 道具分布：Random，為了讓它更平均分布在場中央，所以在 Z 變數有控制他們分別分布在一前、一中、一後。

```
/* shield position */  
shield_1X = (width * 1.9) * rand() / (RAND_MAX + 1.0) - width * 0.95; //shield 1 x座標  
shield_1Z = ((40) * rand() / (RAND_MAX + 1.0)) + 100; //shield 1 z座標
```

- 撿道具：
按下 F 鍵：pick 設為 true，開始撿拾模式
放開 F 鍵：pick 設為 false，結束撿拾模式
撿拾模式(pick==true)時檢查自己和盾牌的距離

```
/* shield collision detection */  
if (pick) { //現在是pick狀態  
    if (shield1_t == 0) { //此盾牌的時間是歸零狀態 (未被撿起)  
        if (abs(shield_1X - meX) < 3 && shield_1Z > (meZ - PICK_LEN) && meZ > shield_1Z) { //檢查是否在夠近的前方  
            shield1_t = SHIELD_TIME; //盾牌保護時間設為8秒(SHIELD_TIME)  
            shield1 = true; //現在是背盾牌1保護的狀態  
        }  
    }  
}
```

- 道具漂浮+影子
 - 道具尚未撿起時，會上下漂浮（星星和盾牌都會，這裡只是以星星為例）。



(a) 下沉，影子拉長



(b) 浮起，影子拉短

```
if(star2){
    glPushMatrix();
    glColor3f(1.0, 0.84313, 0);
    glTranslatef(star_2X, star_2Y, star_2Z); //start位置
    glPushMatrix();
    glTranslatef(0, fly, 0); //漂浮
    glScalef(2, 2, 2);
    glRotatef(90, 0.0, 1.0, 0.0);
    glmDraw(star_model, GLM_SMOOTH);
    glPopMatrix();

    //影子*****
    glPushMatrix();
    glDisable(GL_CULL_FACE);
    glDisable(GL_LIGHT0);
    glDisable(GL_NORMALIZE);
    glDisable(GL_COLOR_MATERIAL);
    glDisable(GL_LIGHTING);
    glColor4f(0.65, 0.53, 0.28, 0);
    glBegin(GL_POLYGON);
    //影子長寬做變化
    glVertex3f(2.0 - shade_cut, -1.99, 0.0);
    glVertex3f(0.0, -1.99, 0.5 - (shade_cut / 4));
    glVertex3f(-2.0 + shade_cut, -1.99, 0.0);
    glVertex3f(0.0, -1.9, -0.5 + (shade_cut / 4));
    glEnd();
    glEnable(GL_CULL_FACE);
    glEnable(GL_LIGHT0);
    glEnable(GL_NORMALIZE);
    glEnable(GL_COLOR_MATERIAL);
    glEnable(GL_LIGHTING);
    glPopMatrix();
    //*****
    glPopMatrix();
}
```

[盾牌道具 詳細說明]

- 繪製：和人物繪製一樣，用長方體貼貼圖而成。
- 撿起後出現在角色前方隨角色移動
- 計時條下會出現 unbeatable time 顯示剩餘保護時間



(a) 未撿起



(b) 撿起後

- Bool shield1：現在是否在被盾牌 1 保護的狀態。
int shield1_t：剩餘保護時間，撿起後設為 8，倒數完後設為-1 以免再次出現。

```
/* 畫盾牌 */
if (unbeat || shield1 || shield2 || shield3) { //正在保護模式前方就出現盾牌
    glPushMatrix();
    glTranslatef(meX, 0, meZ);
    DrawShield(20.0, GL_QUADS);
    glPopMatrix();
}
if (shield1_t == 0) { //盾牌保護時間歸零時顯示在原本位置
    glPushMatrix();
    glTranslatef(shield_1X, 0, shield_1Z);
    glPushMatrix();
    glTranslatef(0, fly, 0);
    DrawShield(20.0, GL_QUADS);
    glPopMatrix();
}
```

- 撿起後出現在角色前方隨角色移動。

```
/* 畫盾牌 */
if (unbeat || shield1 || shield2 || shield3) { //正在保護模式前方就出現盾牌
    glPushMatrix();
    glTranslatef(meX, 0, meZ);
    DrawShield(20.0, GL_QUADS);
    glPopMatrix();
}
```

- 這時移動不會死，因為 shield 倍設為 true，因此呼叫 move()時不會死(die 不會被設為 true)。

```
void move(int m) {
    if (kill == 1 && die == false && !(unbeat || shield1 || shield2 || shield3) && !pass) {
        die = true;
    }
}
```

- 剩餘保護時間

```
void timerFunc(int nTimerID){
    switch (nTimerID){
        case REDISPLAYTIMERID:
            if (time_count > 0 && !pass && !start_menu) {
                time_count--;
                glutPostRedisplay();
                /* shield time countdown */
                if (shield1_t > 0) {
                    shield1_t--;
                    if (shield1_t == 0) {
                        shield1 = false;
                        shield1_t = -1;
                    }
                }
            }
    }
}
```

[星星道具 詳細說明]

- 繪製：load obj file



- 撿取後，遊戲時長加八秒，短暫顯示紅字做提醒



```
/* 拾取星星後會有一小段時間顯示紅字，star_t 初值=0*/
star_t = (star_t+1) % 13;
if (star_t == 12) {
    star_add_time = false;
}
```

```
if (star_add_time) {
    glColor3f(1, 0, 0); //成功吃到星星 文字顏色變紅色做提醒
}
else {
    glColor3f(0, 0, 0);
}
```


五、Results

(a) 起始畫面



(b) 按下 enter，開始遊戲



(c) 撿取星星，加八秒



(d) 撿取盾牌，有八秒不會被殺



(e) 結果 1: 成功通關



(e) 結果 2: 被殺死



六、Feedback

這堂課真的讓我學到了很多。

以前我是不玩電腦遊戲的，所以學期剛開始的時候聽老師介紹那些遊戲的東西有點無感，但是後來開始玩一款 MMORPG 遊戲，才真正實際體會到老師上課說的那些例子，讓我對遊戲本身的架構有更進一步的認識。

在做期末 project 的期間同時要準備其他科的期末考，雖然真的很累，但是報告完很有成就感，能和同組的同學一起寫程式也很開心。