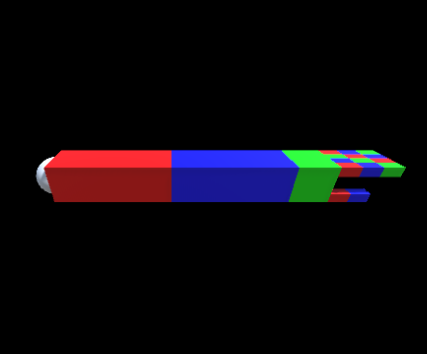
1. Cấu trúc cánh tay
   1. Hình ảnh



* 1. Giải thích
     + Sử dụng các khối Sphere và Cube để tạo cánh tay có đầy đủ 5 ngón theo đúng cấu trúc cây bên trên.
     + Các phần BapTay, CangTay và các đốt ngón tay được tạo bằng khối Cube
     + Các khớp nối CanhTay, Khuy Tay, NgonTay và các Dot2, Dot3 được tạo bằng các khối Sphere
     + Các phần của cánh tay được phân biệt bằng cách tô màu với 3 màu : red, green, blue

1. Cài đặt môi trường
   1. Nguồn sáng:

GLfloat light\_position[] = { 0.0, 20.0, 20.0, 0.0 };

glLightfv(GL\_LIGHT0, GL\_POSITION, light\_position);

glEnable(GL\_LIGHTING);

glEnable(GL\_LIGHT0);

glEnable(GL\_DEPTH\_TEST);

* Nguồn sáng trắng GL\_LIGHT0 tại vị trí (0,20,20)
  1. Khử mặt khuất

glutInitDisplayMode (GLUT\_SINGLE | GLUT\_RGB | GLUT\_DEPTH);

glClear (GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);

* 1. Khung nhìn:

glViewport (0, 0, (GLsizei) w, (GLsizei) h);

glMatrixMode (GL\_PROJECTION);

glLoadIdentity ();

glFrustum (-2.5, 2.5, -2.5, 2.5, 1.5, 50.0);

glMatrixMode (GL\_MODELVIEW);

* Khung nhìn có chiều sâu (xa gần) với kích thước (5x5), gần nhất 1.5, xa nhất 50
  1. Camera

gluLookAt (0.0, 0.0, 6.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0);

* Camera đặt tại +6 trục z, hướng +1 y, nhìn về gốc tọa độ

1. Khởi tạo cánh tay bằng OpenGL
   1. Sử dụng glPushMatrix() và glPopMatrix() để tách biệt các bộ phận

// Canh Tay

glPushMatrix();

...

// Bap Tay

glPushMatrix();

...

glPopMatrix();

// Khuy Tay

glPushMatrix();

...

// Cang Tay

glPushMatrix();

...

glPopMatrix();

// Co Tay

glPushMatrix();

...

// Ban Tay

glPushMatrix();

...

// Ngon Tay

glPushMatrix();

...

// Cube

glPushMatrix(); ...

glPopMatrix();

// Dot2

glPushMatrix();

...

//Cube

glPushMatrix();

...

glPopMatrix();

glPopMatrix();

glPopMatrix();

// Ngon Tay 1

glPushMatrix();

...

// Cube

glPushMatrix(); ...

glPopMatrix();

// Dot2

glPushMatrix();

...

//Cube

...

// Dot3

glPushMatrix();

...

//Cube

glPushMatrix();

...

glPopMatrix();

glPopMatrix();

glPopMatrix();

glPopMatrix();

// Ngon Tay 2 (Tương tự NT1)

// Ngon Tay 3 (Tương tự NT1)

// Ngon Tay 4 (Tương tự NT1)

// Ngon Tay 5 (Tương tự NT1)

glPopMatrix();

glPopMatrix();

glPopMatrix();

glPopMatrix();

* 1. Xây dựng khối và màu vật liệu
     + Khối cube: glutSolidCube(1.0);
     + Khối Sphere: glutSolidSphere(0.5, 64, 64);
     + Phương thức thay đổi màu vật liệu:

void SetMaterialColor(GLfloat mat\_diffuse[4]){

GLfloat mat\_specular[] = { 1.0, 1.0, 1.0, 1.0 };

GLfloat mat\_ambient[] = { 0.0, 0.0, 0.0, 1.0 };

GLfloat mat\_shininess[] = { 50.0 };

glMaterialfv(GL\_FRONT\_AND\_BACK, GL\_SPECULAR, mat\_specular);

glMaterialfv(GL\_FRONT\_AND\_BACK, GL\_DIFFUSE, mat\_diffuse);

glMaterialfv(GL\_FRONT\_AND\_BACK, GL\_AMBIENT, mat\_ambient);

glMaterialfv(GL\_FRONT\_AND\_BACK, GL\_SHININESS, mat\_shininess);

}

* Thiết lặp màu mặc định specular : trắng, ambient đen và diffuse là 1 biến truyền vào
* Gán màu cho mặt trước và sau của vật liệu : GL\_FRONT\_AND\_BACK
  + - Khởi tạo màu cơ bản:

GLfloat red[] = { 1.0, 0, 0, 1.0 };

GLfloat green[] = { 0, 1.0, 0, 1.0 };

GLfloat blue[] = { 0, 0, 1.0, 1.0 };

* + - Gán màu cho vật liệu: SetMaterialColor(red); SetMaterialColor(green); SetMaterialColor(blue);
    - Sử dụng các phép biến đổi cơ bản để biến đổi các khối: glTranslatef(z, y, z); glScalef(y, x, z); glRotatef(angle, x, y, z);

1. Tạo hoạt ảnh (animation)
   1. Khởi tạo các biến tương ứng các góc quay tại các khớp trên cánh tay theo 3 trục x, y, z

float xspCaT = xspKhT = xspCoT = xspNT = xspD2NT = xspNT1 = xspD2NT1 =

xspD3NT1 = xspNT2 = xspD2NT2 = xspD3NT2 = xspNT3 = xspD2NT3 = xspD3NT3 = xspNT4 = xspD2NT4 = xspD3NT4 = 0;

float yspCaT = yspKhT = yspCoT = yspNT = yspD2NT = yspNT1 = yspD2NT1 =

yspD3NT1 = yspNT2 = yspD2NT2 = yspD3NT2 = yspNT3 = yspD2NT3 = yspD3NT3 = yspNT4 = yspD2NT4 = yspD3NT4 = 0;

float zspCaT = zspKhT = zspCoT = zspNT = zspD2NT = zspNT1 = zspD2NT1 =

zspD3NT1 = zspNT2 = zspD2NT2 = zspD3NT2 = zspNT3 = zspD2NT3 = zspD3NT3 = zspNT4 = zspD2NT4 = zspD3NT4 = 0;

* 1. Bắt sự kiện với bàn phím

void keyboard(unsigned char key, int x, int y){

// kich hoat default

if(key == '0')...

// kich hoat dong tac 1

if(key == '1')...

// kich hoat dong tac 2

if(key == '2')...

// kich hoat dong tac 3

if(key == '3')...

// DT1 --> nam tay

if(key == 'a')...

// DT2 --> bung tay

if(key == 's')...

// DT3 --> vay tay

if(key == 'd')...

}

glutKeyboardFunc(keyboard);

* Giải thích: Lắng nghe sự kiện nhấn phím trên bàn phím:
  + Nhấn ‘0’: Kích hoạt trạng thái mặc định:

DT1 = false;

DT2 = false;

DT3 = false;

glutPostRedisplay();

* + Nhấn ‘1’: Kích hoạt động tác 1: Nắm tay

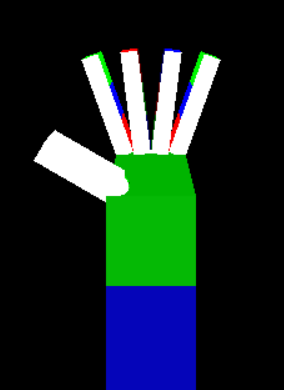
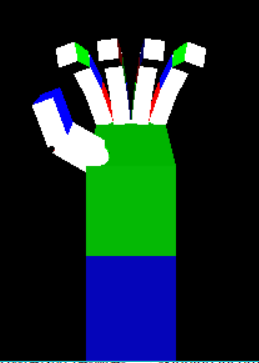
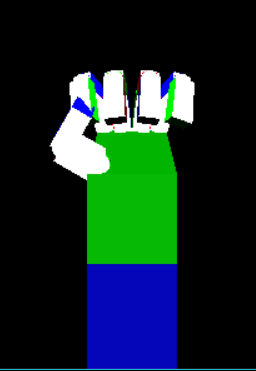
DT1 = true;

DT2 = false;

DT3 = false;

glutPostRedisplay();

* + Tương tự khi nhấn ’2’ và ‘3’ để kích hoạt 2 động tác còn lại
  + Nhấn ‘a’: Để thực hiện Nắm tay: glutIdleFunc(NamTay);
  + Nhấn ‘s’: Để thực hiện Búng tay: glutIdleFunc(BungTay);
  + Nhấn ‘d’: Để thực hiện Vẫy tay: glutIdleFunc(VayTay);
  1. Động tác 1: Nắm tay
     + Hình ảnh

* + - Thiết lập trạng thái mặc định cho động tác 1

if(DT1){

glTranslatef(3.0, -6.0, 6.0);

glRotatef(90, 0.0, 1.0, 0.0);

}

if(DT1) glRotatef(90, 0.0, 0.0, 1.0);

if(DT1) glRotatef(180, 1.0, 0.0, 0.0);

if(DT1) glRotatef(-60, 0.0, 1.0, 0.0);

if(DT1) glRotatef(-20, 0.0, 1.0, 0.0);

if(DT1) glRotatef(-7.5, 0.0, 1.0, 0.0);

if(DT1) glRotatef(7.5, 0.0, 1.0, 0.0);

if(DT1) glRotatef(20, 0.0, 1.0, 0.0);

* Giải thích:
  + Từ trạng thái mặc định di chuyển cánh tay gần về phía Camera
  + Thực hiện quay CanhTay, KhuyTay, CoTay hướng về phía Camera, các ngón tay xòe ra
    - Động tác nắm tay

if(DT1) glRotatef(yspD2NT, 0.0, 1.0, 0.0);

if(DT1) glRotatef(zspD2NT1, 0.0, 0.0, 1.0);

if(DT1) glRotatef(zspD2NT2, 0.0, 0.0, 1.0);

if(DT1) glRotatef(zspD2NT3, 0.0, 0.0, 1.0);

if(DT1) glRotatef(zspD2NT4, 0.0, 0.0, 1.0);

if(DT1) glRotatef(zspD3NT1, 0.0, 0.0, 1.0);

if(DT1) glRotatef(zspD3NT2, 0.0, 0.0, 1.0);

if(DT1) glRotatef(zspD3NT3, 0.0, 0.0, 1.0);

if(DT1) glRotatef(zspD3NT4, 0.0, 0.0, 1.0);

int sDT1 = 0;

void NamTay(void){

sDT1++;

yspD2NT += 0.2;

zspD2NT1 -= 0.1;

zspD2NT2 -= 0.1;

zspD2NT3 -= 0.1;

zspD2NT4 -= 0.1;

zspD3NT1 -= 0.2;

zspD3NT2 -= 0.2;

zspD3NT3 -= 0.2;

zspD3NT4 -= 0.2;

if(sDT1 > 450) return;

glutPostRedisplay();

}

* Giải thích:
  + Thực hiện 450 bước để nắm tay
  + Các chuyển động của 5 ngón tay thực hiện nắm
  + Đốt 2 Ngón tay cái gập vào chiều y+ 90 độ, mỗi bước +0.2 độ
  + Đốt 2 của 4 Ngón tay (1234) gập vào chiều z- 45 độ, mỗi bước -0.1 độ
  + Đốt 4 của 4 Ngón tay (1234) gập vào chiều z- 90 độ, mỗi bước -0.2 độ
  1. Động tác 2: Búng tay
  2. Động tác 3: Vẫy tay