**Objek Babi**

Projek Individu yang saya gunakan adalah objek Babi. Objek ini menggunakan 3 objek quadratik yaitu 2 kotak (1 badan dan 1 kepala), 4 oval (untuk ke-4 kaki), dan 2 lingkaran (untuk ke-2 mata). Untuk animasi hanya menggunakan rotateX() dan rotateY(). Berikut susunan program dalam Objek Babi.

1. **Kubus**

Kubus ini dibuat untuk bagian kepala dan badan babi. Badan babi dibuat lebih panjang sehingga terlihat balok. Berikut code kepala dan badan babi:

**Badan Babi:**

private void CreateCubeVertices()

{

float posX = 0.0f;

float posY = 0.0f;

float posZ = 0.6f;

float boxlength = 1.0f;

float boxwidth = 0.4f;

//titik 1

\_box\_vertices[0] = posX - boxlength / 2.0f;

\_box\_vertices[1] = posY + boxwidth / 2.0f;

\_box\_vertices[2] = posZ - boxwidth / 2.0f;

//titik 2

\_box\_vertices[3] = posX + boxlength / 2.0f;

\_box\_vertices[4] = posY + boxwidth / 2.0f;

\_box\_vertices[5] = posZ - boxwidth / 2.0f;

//titik 3

\_box\_vertices[6] = posX - boxlength / 2.0f;

\_box\_vertices[7] = posY - boxwidth / 2.0f;

\_box\_vertices[8] = posZ - boxwidth / 2.0f;

//titik 4

\_box\_vertices[9] = posX + boxlength / 2.0f;

\_box\_vertices[10] = posY - boxwidth / 2.0f;

\_box\_vertices[11] = posZ - boxwidth / 2.0f;

//titik 5

\_box\_vertices[12] = posX - boxlength / 2.0f;

\_box\_vertices[13] = posY + boxwidth / 2.0f;

\_box\_vertices[14] = posZ + boxwidth / 2.0f;

//titik 6

\_box\_vertices[15] = posX + boxlength / 2.0f;

\_box\_vertices[16] = posY + boxwidth / 2.0f;

\_box\_vertices[17] = posZ + boxwidth / 2.0f;

//titik 7

\_box\_vertices[18] = posX - boxlength / 2.0f;

\_box\_vertices[19] = posY - boxwidth / 2.0f;

\_box\_vertices[20] = posZ + boxwidth / 2.0f;

//titik 8

\_box\_vertices[21] = posX + boxlength / 2.0f;

\_box\_vertices[22] = posY - boxwidth / 2.0f;

\_box\_vertices[23] = posZ + boxwidth / 2.0f;

}

Karena memanjang, titik kubus dibuat 2 variable yang berbeda. Boxlength ditaruh di titik X yang artinya X tersebut akan memajang sesuai input pada variable Boxlength. BoxWidth berguna untuk melebarkan garis Y dan Z. Hasil gambar seperti dibawah ini:

**Kepala Babi:**

Pada function kepala babi, tidak berbeda jauh dengan function untuk membuat badan babi. Hanya saja perbedaannya Boxlength = BoxWidth sehingga akan menjadi sebuah balok. Berikut variable yang telah dibuat.

float posX = 0.3f;

float posY = 0.1f;

float posZ = 0.6f;

float boxlength = 0.4f;

float boxwidth = 0.4f;

Karena codenya sama seperti bagian badan babi, maka bagian kepala yang berbentuk kubus nilai variablenya dibuat sama. Hasil Gambar akan seperti ini:

1. **Ellipsoid**

Ellipsoid pada babi ini digunakan sebagai kaki babi yang berjumlah 4. Pada dasarnya semua function untuk membuat kaki ini adalah sama, hanya saja titik penempatan ke-4 kaki tersebut berbeda-beda. Berikut adalah function membuat per kakinya:

public void createCircleVertices()

{

for (float u = -pi; u <= pi; u += pi / 30)

{

for (float v = -pi / 2; v < pi / 2; v += pi / 30)

{

\_ellipsoid\_vertices[\_ellipsoid\_index \* 3] = PosX + radX \* (float)Math.Cos(v) \* (float)Math.Cos(u); //x

\_ellipsoid\_vertices[\_ellipsoid\_index \* 3 + 1] = PosY + radY \* (float)Math.Cos(v) \* (float)Math.Sin(u); //y

\_ellipsoid\_vertices[\_ellipsoid\_index \* 3 + 2] = PosZ + radX \* (float)Math.Sin(v); //z

\_ellipsoid\_index++;

}

}

}

Karena seluruh function pembuatan kaki sama, dalam function tersebut tentu memiliki nilai variable yang berbeda-beda untuk titik X, Y, Z-nya. Selain itu, ellipsoid yang digunakan berbentuk seperti oval, karena itu ada tambahan variable baru yaitu radY yang berfungsi untuk jari-jari sumbu Y sehingga akan berbentuk melonjong / oval. Berikut variable-variable yang digunakan dalam membuat kaki babi:

//kaki 1

float PosX = -0.3f;

float PosY = -0.35f;

float PosZ = 0.43f;

float radX = 0.07f;

float radY = 0.15f;

float pi = 3.14159265359f;

//kaki 2

float PosX = -0.3f;

float PosY = -0.35f;

float PosZ = 0.73f;

float radX = 0.07f;

float radY = 0.15f;

float pi = 3.14159265359f;

//kaki 3

float PosX = 0.33f;

float PosY = -0.35f;

float PosZ = 0.43f;

float radX = 0.07f;

float radY = 0.15f;

float pi = 3.14159265359f;

//kaki 4

float PosX = 0.33f;

float PosY = -0.35f;

float PosZ = 0.73f;

float radX = 0.07f;

float radY = 0.15f;

float pi = 3.14159265359f;

Terlihat bahwa masing-masing posX, posY, posZ memiliki nilai sendiri-sendiri dan apabila digambar akan terlihat seperti berikut:

1. **Bola**

Bola digunakan untuk kedua mata babi. Mata babi tersebut menggunakan objek quadratik ellipsoid seperti kakinya, hanya saja jari-jari x = jari jari y. Berikut function membuat mata babi.

public void createEye() {

for (float u = -pi; u <= pi; u += pi / 30)

{

for (float v = -pi / 2; v < pi / 2; v += pi / 30)

{

\_ellipsoid\_vertices4[\_ellipsoid\_index4 \* 3] = PosX + rad \* (float)Math.Cos(v) \* (float)Math.Cos(u); //x

\_ellipsoid\_vertices4[\_ellipsoid\_index4 \* 3 + 1] = PosY + rad \* (float)Math.Cos(v) \* (float)Math.Sin(u); //y

\_ellipsoid\_vertices4[\_ellipsoid\_index4 \* 3 + 2] = PosZ + rad \* (float)Math.Sin(v); //z

\_ellipsoid\_index4++;

}

}

}

Karena function kedua mata sama, maka yang akan berbeda adalah posisi X,Y, dan Z-nya. Berikut adalah variable yang digunakan untuk kedua bola mata.

//Mata 1

float PosX = 0.5f;

float PosY = 0.2f;

float PosZ = 0.7f;

float rad = 0.05f;

float pi = 3.14159265359f;

//Mata 2

float PosX = 0.5f;

float PosY = 0.2f;

float PosZ = 0.5f;

float rad = 0.05f;

float pi = 3.14159265359f;

Terlihat bahwa variable dari kedua bola mata tersebut hampir mirip, hanya saja pada PosZ, bola mata hanya bergeser untuk ke kiri atau ke kanan. Apabila digambar hasilnya akan seperti dibawah ini:

1. **Bezier**

Pada objek babi ini, saya menggunakan garis bezier untuk bagian ekor babi. Bezier pada objek ini adalah menggunakan 3 vektor (x,y,z) yang dimana 3 vektor tersebut merupakan vektor 3 dimensi. Berikut code yang melakukan perhitungan serta menggambar garis beziernya:

Vector3 setBezier(Vector3 p1, Vector3 p2, Vector3 p3, Vector3 p4, float t)

{

Vector3 p;

float a1 = (float)Math.Pow((1 - t), 3);

float a2 = (float)Math.Pow((1 - t), 2) \* 3 \* t;

float a3 = 3 \* t \* t \* (1 - t);

float a4 = t \* t \* t;

p.X = a1 \* p1.X + a2 \* p2.X + a3 \* p3.X + a4 \* p4.X;

p.Y = a1 \* p1.Y + a2 \* p2.Y + a3 \* p3.Y + a4 \* p4.Y;

p.Z = a1 \* p1.Z + a2 \* p2.Z + a3 \* p3.Z + a4 \* p4.Z;

return p;

}

Code diatas merupakan function yang dimana isi function tersebut digunakan untuk meng-set titik yang akan digariskan. Untuk menggariskannya, berikut code dibawah ini yang menjalankan garis tersebut.

if (calculate == 0)

{

float[] \_x = new float[5];

float[] \_y = new float[5];

float[] \_z = new float[5];

\_x[0] = (float)-0.5175;

\_y[0] = (float)-0.01;

\_z[0] = (float)0;

\_x[1] = (float)-0.83;

\_y[1] = (float)-0.16;

\_z[1] = (float)0;

\_x[2] = (float)-0.6375;

\_y[2] = (float)0.145;

\_z[2] = (float)0;

\_x[3] = (float)-0.5125;

\_y[3] = (float)-0.235;

\_z[3] = (float)0;

//scale();

\_vertices\_bezier\_temporary[0].X = \_x[0];

\_vertices\_bezier\_temporary[0].Y = \_y[0];

\_vertices\_bezier\_temporary[0].Z = \_z[0];

\_vertices\_bezier\_temporary[1].X = \_x[1];

\_vertices\_bezier\_temporary[1].Y = \_y[1];

\_vertices\_bezier\_temporary[1].Z = \_z[1];

\_vertices\_bezier\_temporary[2].X = \_x[2];

\_vertices\_bezier\_temporary[2].Y = \_y[2];

\_vertices\_bezier\_temporary[2].Z = \_z[2];

\_vertices\_bezier\_temporary[3].X = \_x[3];

\_vertices\_bezier\_temporary[3].Y = \_y[3];

\_vertices\_bezier\_temporary[3].Z = \_z[3];

for (int index = 0; index < 4; index++)

{

\_vertices\_line[index \* 3] = \_x[index];

\_vertices\_line[index \* 3 + 1] = \_y[index];

\_vertices\_line[index \* 3 + 1] = \_z[index];

}

\_vertices\_bezier[i] = \_vertices\_line[0];

i++;

//vertex kedua (y)

\_vertices\_bezier[i] = \_vertices\_line[1];

i++;

//vertex ketiga (z)

\_vertices\_bezier[i] = \_vertices\_line[2];

i++;

//t += =>> nilainya bisa diganti-ganti

\_vertice\_bezier\_count = 1;

for (float t = 0.0f; t <= 1.0f; t += 0.01f)

{

Vector3 P = setBezier(\_vertices\_bezier\_temporary[0], \_vertices\_bezier\_temporary[1],

\_vertices\_bezier\_temporary[2], \_vertices\_bezier\_temporary[3], t);

\_vertices\_bezier[i] = P.X;

i++;

\_vertices\_bezier[i] = P.Y;

i++;

\_vertices\_bezier[i] = P.Z;

i++;

\_vertice\_bezier\_count++;

}

calculate = 0;

}

Code diatas ini ditempatkan pada function onranderframe(); yang dimana function tersebut akan menggambarkan garis bezier di layar. Sehingga keseluruhannya akan menghasilkan gambar berikut.

**Daftar Animasi per Objek**

Berikut tabel daftar animasi per Objek

|  |  |
| --- | --- |
| **Objek** | **Animasi** |
| Badan Babi | RotateY(); |
| Kepala Babi | RotateY(); |
| Kaki 1 | RotateY(); |
| Kaki 2 | RotateY(); |
| Kaki 3 | RotateY(); |
| Kaki 4 | RotateY(); |
| Mata 1 | RotateY(); |
| Mata 2 | RotateY(); |
| Bezier | RotateY(); |