OpenGl

Ex: Animated circle

#include <gl/glut.h>

#include <math.h>

struct Point {

   GLint x;

   GLint y;

};

void init() {

   glClearColor(1.0, 1.0, 1.0, 0.0);

   glColor3f(0.0f, 0.0f, 0.0f);

   glPointSize(1.0f);

   gluOrtho2D(0.0f, 640.0f, 0.0f, 480.0f);

}

void draw\_circle(Point pC, GLfloat radius) {

   GLfloat step = 1/radius;

   GLfloat x, y;

   for(GLfloat theta = 0; theta <= 360; theta += step) {

      x = pC.x + (radius \* cos(theta));

      y = pC.y + (radius \* sin(theta));

      glVertex2i(x, y);

   }

}

Point pCenter = {320, 240};

float dRadius = 1;

float radius[3] = {

   0.0f,

   60.0f,

   120.0f

};

void display(void) {

   glClear(GL\_COLOR\_BUFFER\_BIT);

   glBegin(GL\_POINTS);

      for(int i = 0; i < 3; i++) {

         draw\_circle(pCenter, radius[i]);

         radius[i]+=dRadius;

         if(radius[i] > 180)

            radius[i] = 0;

      }

   glEnd();

   glFlush();

}

void Timer(int value) {

   glutTimerFunc(33, Timer, 0);

   glutPostRedisplay();

}

int main(int argc, char \*\*argv) {

   glutInit(&argc, argv);

   glutInitDisplayMode(GLUT\_SINGLE|GLUT\_RGB);

   glutInitWindowPosition(200, 200);

   glutInitWindowSize(640, 480);

   glutCreateWindow("Square");

   glutDisplayFunc(display);

   init();

   Timer(0);

   glutMainLoop();

   return 0;

}

Output:

