

Ventana de dibujo 2D

E. Camahort

Planteamiento

Esta ficha explica cómo establecer las coordenadas de una ventana de dibujo 2D. Dichas coordenadas fijan la ventana del mundo real cuyo contenido será visualizado. Se utiliza para el dibujo y visualización de primitivas gráficas 2D como cuadrados, rectángulos, círculos e imágenes.

Solución

La orden OpenGL que permite establecer una ventana de dibujo 2D es la orden `gluOrtho2D`. Su formato es el siguiente:

```
void gluOrtho2D(GLdouble izqda, GLdouble drcha, GLdouble
abajo, GLdouble arriba);
```

donde los parámetros especifican las rectas de recortado:

- `izqda` y `drcha` son las coordenadas de las rectas de recortado verticales izquierdo y derecho, y
- `abajo` y `arriba` son las coordenadas de las rectas de recortado horizontales inferior y superior.

Al ejecutarse, esta orden crea una matriz de proyección de coordenadas 2D y la multiplica por la matriz de proyección actual. Si no existen otras transformaciones acumuladas en la matriz de proyección actual, la nueva matriz transforma los objetos 2D de forma que los puntos (`izqda`, `abajo`) y (`drcha`, `arriba`) aparecen respectivamente como $(-1, -1)$ y $(1, 1)$ después de aplicar la transformación. La función `gluOrtho2D` es equivalente a llamar a `glOrtho` con los parámetros `near` y `far` iguales a -1 y 1 , respectivamente.

Ejemplo

Las imágenes de la figura 2.1 muestran el mismo triángulo blanco dibujado en dos ventanas de dibujo 2D distintas (ver *coord2d.c*). En el primer caso las rectas de recortado pasan por los puntos $(-1, -1)$ y $(1, 1)$. En el segundo caso las rectas de recortado pasan por los puntos $(-1, -0.5)$ y $(0.5, 0.5)$. Nótese que con el cambio de ventana de dibujo el triángulo aparece deformado y situado más cerca del borde derecho de la imagen.

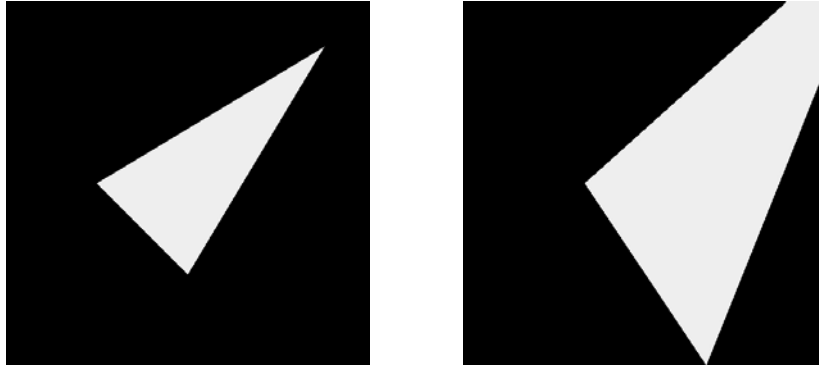


Figura 2.1: Resultado obtenido utilizando distintos parámetros con `gluOrtho2D`.

Los dos fragmentos de código que se encuentran a continuación contienen las órdenes necesarias para obtener las dos imágenes utilizando OpenGL.

```
glMatrixMode(GL_MODELVIEW);
glLoadIdentity();

glMatrixMode(GL_PROJECTION);
glLoadIdentity();
gluOrtho2D(-1, 1, -1, 1);
glBegin(GL_TRIANGLES);
    glVertex2f(-0.5, 0);
    glVertex2f(0, -0.5);
    glVertex2f(0.75, 0.75);
glEnd();

glMatrixMode(GL_PROJECTION);
glLoadIdentity();
```

```
gluOrtho2D(-1, 0.5, -0.5, 0.5);
glBegin(GL_TRIANGLES);
    glVertex2f(-0.5, 0);
    glVertex2f(0, -0.5);
    glVertex2f(0.75, 0.75);
glEnd();
```

Fichas relacionadas

- «La proyección paralela», página 65.
- «La proyección perspectiva», página 69.

Bibliografía

- Capítulo 3 de Mason Woo, Jackie Neider, Tom Davis, Dave Shreiner, *OpenGL programming guide: the official guide to learning OpenGL, (Version 1.2)*, 1999.