Modelación de Sistemas Multiagentes con Gráficas Computacionales

Actividad 4: Operadores Geométricos 3D

TC2009B

Grupo 501

Alumnos

Rodrigo López Guerra

Maestros

Luciano García Bañuelos

Iván Olmos Pineda

Semestre

5° Semestre

Introducción

En los gráficos 3D, las transformaciones geométricas como **traslación**, **rotación** y **escalado** son fundamentales para manipular objetos dentro del espacio tridimensional. Estas operaciones se representan matemáticamente mediante matrices y se aplican a los vértices de los objetos para cambiar su posición, orientación o tamaño.

El presente documento describe cómo implementar estas transformaciones utilizando matrices de 4x4 en un entorno de gráficos 3D basado en Python, empleando bibliotecas como **NumPy** para los cálculos matriciales y **OpenGL** para la renderización. Se explica detalladamente cómo estas transformaciones permiten mover, escalar y rotar objetos de forma acumulativa, logrando combinaciones complejas de animación y posicionamiento en el espacio tridimensional.

Traslación

La traslación es el proceso de mover un objeto en el espacio tridimensional, desplazándose a lo largo de los ejes X, Y y Z. Una matriz de traslación es una matriz de 4x4 donde los valores de traslación (tx, ty, tz) se colocan en la última columna de la matriz.

```
[ 1   0   0   tx ]
[ 0   1   0   ty ]
[ 0   0   1   tz ]
[ 0   0   0   1  ]
```

- La matriz de traslación se multiplica por la matriz acumulada (`self.M`), que contiene todas las transformaciones aplicadas previamente.
- La multiplicación asegura que la traslación se acumule con otras transformaciones ya realizadas, como rotaciones o escalados.
- Cada vértice del objeto es transformado por esta matriz, sumando los desplazamientos a sus coordenadas actuales.
- Por ejemplo, un punto `(x, y, z)` se convierte en `(x + tx, y + ty, z + tz)`.

Escalado

El escalado cambia el tamaño del objeto en el espacio tridimensional, haciendo que se haga más grande o más pequeño en cada eje (X, Y y Z). Una matriz de escalado es una matriz de 4x4 con los factores de escala (`sx`, `sy`, `sz`) colocados en la diagonal principal.

```
[ sx   0   0  0 ]
[ 0   sy   0  0 ]
[ 0   0   sz  0 ]
[ 0   0   0   1 ]
```

- La matriz de escalado se multiplica por la matriz acumulada (`self.M`), permitiendo que el escalado se combine con otras transformaciones.
- Cada vértice del objeto es multiplicado por el factor correspondiente al eje, por ejemplo, `(x, y, z)` se convierte en `(x * sx, y * sy, z * sz)`.
- Si los factores de escalado son mayores a 1, el objeto se agranda.
- Si los factores están entre 0 y 1, el objeto se encoge.
- Esto permite cambiar el tamaño del objeto sin alterar su forma relativa.

Rotación en Eje Específico

La rotación hace que el objeto gire alrededor de un eje específico (X, Y o Z) o alrededor de un eje arbitrario definido por el usuario.

Para cada eje, se definió una matriz de rotación de 4x4 utilizando funciones trigonométricas (`sin` y `cos`), ya que la rotación en un espacio tridimensional implica una relación entre los ángulos y las coordenadas.

Por ejemplo, la matriz de rotación alrededor del eje Z tiene la forma:

```
[ cos(θ) -sin(θ)   0  0 ]
[ sin(θ)  cos(θ)   0  0 ]
[   0       0      1  0 ]
[   0       0      0  1 ]
```

- Aquí, `θ` es el ángulo de rotación en radianes.
- Para rotaciones alrededor de los ejes X y Y, se usan diferentes combinaciones de `cos` y `sin` para modificar las filas y columnas correspondientes a los ejes afectados.
- La matriz de rotación se multiplica por la matriz acumulada (`self.M`).
- Esto permite que la rotación se combine con otras transformaciones en el orden correcto.
- Cada vértice del objeto se mueve en un arco alrededor del eje de rotación.
- Por ejemplo, un punto `(x, y, z)` que gira 90° alrededor del eje Z cambiará a `(-y, x, z)`.

Rotación en Eje Arbitrario

Se usó una fórmula matemática general para rotar un objeto alrededor de cualquier eje definido por un vector (x, y, z).

Esta fórmula incluye:

- Normalización del vector (x, y, z) para asegurarse de que su longitud sea 1.
- Cálculo de una matriz de rotación usando trigonometría y álgebra lineal:

```
[ t*x*x + c   t*x*y - s*z   t*x*z + s*y   0 ]
[ t*x*y + s*z t*y*y + c     t*y*z - s*x   0 ]
[ t*x*z - s*y t*y*z + s*x   t*z*z + c     0 ]
[ 0           0             0             1 ]
```

- `t = 1 - cos(θ)`
- `c = cos(θ)`
- `s = sin(θ)`
- `(x, y, z)` es el vector del eje de rotación.

- La matriz de rotación se multiplica por la matriz acumulada (`self.M`).
- Esto permite que la rotación se combine con otras transformaciones en el orden correcto.
- Cada vértice del objeto se mueve en un arco alrededor del eje de rotación.
- Por ejemplo, un punto `(x, y, z)` que gira 90° alrededor del eje Z cambiará a `(-y, x, z)`.

Conclusiones

Las transformaciones geométricas como la **traslación**, **rotación** y **escalado** son la base de los gráficos 3D y permiten controlar de manera precisa el comportamiento y la interacción de los objetos en un espacio tridimensional. A través del uso de matrices, estas operaciones se combinan eficientemente para crear movimientos, animaciones y estructuras complejas.

La implementación mediante matrices de 4x4 asegura que las transformaciones puedan acumularse de forma ordenada, permitiendo flexibilidad y control sobre los objetos. Además, su representación en código usando bibliotecas como **NumPy** y **OpenGL** proporciona una forma robusta y eficiente de integrar cálculos matemáticos y renderización gráfica.

Comprender y aplicar estas técnicas no solo es esencial para el desarrollo de gráficos 3D, sino también para campos como la simulación, la realidad virtual y los videojuegos, donde la manipulación precisa de objetos en un espacio tridimensional es fundamental para crear experiencias inmersivas y dinámicas.