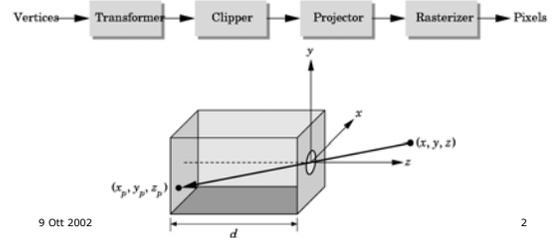


Costruzione di Interfacce Lezione 5 Trasformazioni Affini

cignoni@iei.pi.cnr.it
<http://vcg.iei.pi.cnr.it/~cignoni>

Cambi di Sistemi di riferimento

- Il primo step della pipeline di rendering è quello di trasformare la scena nel sistema di riferimento della camera

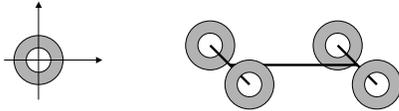


9 Ott 2002

2

Object Frame

- Perché ogni oggetto ha il suo sistema di riferimento?
 - Uso Multiplo di uno stesso oggetto
 - Posizione parametrica



9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

3

Coordinate Omogenee

- Si dice che un punto P è rappresentato dalla matrice colonna \mathbf{p}

$$\mathbf{p} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ 1 \end{bmatrix}$$

- Un vettore w è rappresentato dalla matrice colonna \mathbf{a}

$$\mathbf{a} = \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ 0 \end{bmatrix}$$

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

4

Trasformazioni Affini

- Notare che se \mathbf{u} è un vettore solo 9 elementi di \mathbf{A} sono usati nella trasformazione

$$\mathbf{A}\mathbf{u} = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} & \alpha_{14} \\ \alpha_{21} & \alpha_{22} & \alpha_{23} & \alpha_{24} \\ \alpha_{31} & \alpha_{32} & \alpha_{33} & \alpha_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ 0 \end{bmatrix}$$

- La quarta colonna corrisponde alla quarta riga della matrice di cambiamento di frame, che conteneva il nuovo punto di origine del frame (che chiaramente non serve se si parla di vettori)

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

5

Traslazione

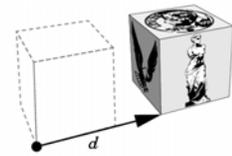
- modifica i punti di un frame sommando a tutti i punti un vettore di spostamento \mathbf{d}

$$\mathbf{P}' = \mathbf{P} + \mathbf{d}$$

$$\mathbf{p}' = \mathbf{p} + \mathbf{d}$$



(a)



(b)

9 Ott 2002

6

Traslazione

$$\mathbf{p}' = \mathbf{p} + \mathbf{d} \quad \mathbf{p} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad \mathbf{p}' = \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} \quad \mathbf{d} = \begin{bmatrix} \alpha_x \\ \alpha_y \\ \alpha_z \\ 0 \end{bmatrix}$$

$$\mathbf{p}' = \mathbf{T}\mathbf{p} \quad \mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & \alpha_x \\ 0 & 1 & 0 & \alpha_y \\ 0 & 0 & 1 & \alpha_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

7

Traslazione

$$\mathbf{T}(\alpha_x, \alpha_y, \alpha_z) = \begin{bmatrix} 1 & 0 & 0 & \alpha_x \\ 0 & 1 & 0 & \alpha_y \\ 0 & 0 & 1 & \alpha_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}^{-1}(\alpha_x, \alpha_y, \alpha_z) = \mathbf{T}(-\alpha_x, -\alpha_y, -\alpha_z) = \begin{bmatrix} 1 & 0 & 0 & -\alpha_x \\ 0 & 1 & 0 & -\alpha_y \\ 0 & 0 & 1 & -\alpha_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

9 Ott 2002

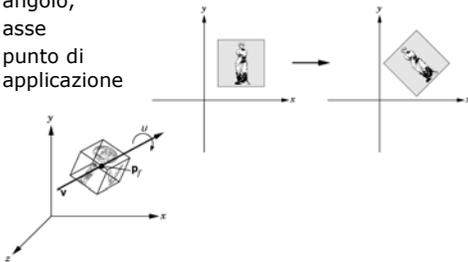
Costruzione di Interfacce - Paolo Cignoni

8

Rotazione

❖ Di una rotazione si deve specificare

- ❖ angolo,
- ❖ asse
- ❖ punto di applicazione



9 Ott 2002

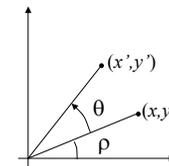
Costruzione di Interfacce - Paolo Cignoni

9

Rotazione

- ❖ Caso semplice asse z, intorno all'origine, di un angolo θ
- ❖ Possiamo considerare il problema in 2d

$$\begin{aligned} x &= \rho \cos \phi \\ y &= \rho \sin \phi \\ x' &= \rho \cos(\phi + \theta) \\ y' &= \rho \sin(\phi + \theta) \end{aligned}$$



9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

10

Rotazione

$$\begin{aligned} x &= \rho \cos \phi \\ y &= \rho \sin \phi \\ x' &= \rho \cos \phi \cos \theta - \rho \sin \phi \sin \theta = x \cos \theta - y \sin \theta \\ y' &= \rho \cos \phi \sin \theta + \rho \sin \phi \cos \theta = x \sin \theta + y \cos \theta \end{aligned}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

11

Rotazione

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \mathbf{R}(\theta)_z \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \cos \theta - y \sin \theta \\ x \sin \theta + y \cos \theta \\ z \\ 1 \end{bmatrix}$$

$$\mathbf{R}_z(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

12

Rotazioni

$$\mathbf{R}_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{R}_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

13

Rotazione inversa

- Le matrici di rotazione viste finora sono invertibili facilmente.

$$\mathbf{R}^{-1}(\theta) = \mathbf{R}(-\theta)$$

$$\sin(-\theta) = -\sin(\theta)$$

$$\cos(-\theta) = \cos(\theta)$$

- Quindi basta trasporre...

$$\mathbf{R}^{-1}(\theta) = \mathbf{R}^T(\theta)$$

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

14

Rotazioni

- Finora abbiamo visto solo rotazioni intorno all'origine e lungo gli assi.
- Una rotazione arbitraria si ottiene componendo piu' rotazioni e traslazioni

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

15

Scalatura

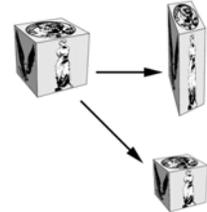
- Non rigida
- Non uniforme lungo gli assi
- Solo centrata all'origine

$$x' = \beta_x x$$

$$y' = \beta_y y$$

$$z' = \beta_z z$$

$$\mathbf{T}(\beta_x, \beta_y, \beta_z) = \begin{bmatrix} \beta_x & 0 & 0 & 0 \\ 0 & \beta_y & 0 & 0 \\ 0 & 0 & \beta_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

16

Composizione di Trasformazioni

- Le trasformazioni sono matrici
- L'applicazione di trasformazione è la moltiplicazione di una matrice per un vettore.
- L'applicazione di una sequenza di trasformazioni ad un punto corrisponde ad una sequenza di moltiplicazioni di di matrici per vettori
- Associatività

$$\mathbf{q} = \mathbf{CBAp}$$

$$\mathbf{q} = (\mathbf{C}(\mathbf{B}(\mathbf{A}\mathbf{p})))$$

$$\mathbf{q} = (\mathbf{CBA})\mathbf{p}$$



9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

17

Composizione di trasformazioni

- Conviene se si deve trasformare un solo punto $\mathbf{q} = (\mathbf{C}(\mathbf{B}(\mathbf{A}\mathbf{p})))$
- Conviene se si deve trasformare molti punti $\mathbf{q} = (\mathbf{CBA})\mathbf{p}$



9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

18

Composizione di Trasformazioni

- ❖ La moltiplicazione tra matrici NON è commutativa
- ❖ Quindi l'ordine delle trasformazioni È importante

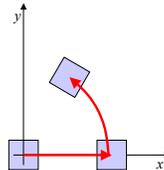
9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

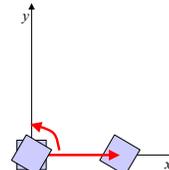
19

Composizione di Trasformazioni

$$q = RTp$$



$$q = TRp$$



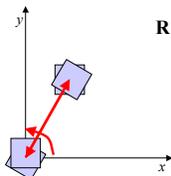
9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

20

Rotazione intorno ad un punto

- ❖ Si ottiene spostando tutto il sistema di riferimento nel punto, facendo la rotazione e rimettendo tutto a posto



$$R_p = T(p)R_0T(-p)$$

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

21

Rotazione intorno ad un asse

- ❖ L'idea e' quella di fare :
 - ❖ due rotazioni che portino l'asse di rotazione specificato a coincidere con l'asse z,
 - ❖ la rotazione lungo quell'asse di quanto specificato
 - ❖ rimettere tutto a posto
- $$R = R_x(-\theta_x)R_y(-\theta_y)R_z(\theta_z)R_y(\theta_y)R_x(\theta_x)$$
- ❖ trovare θ_y, θ_x non è semplice...

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

22

Shearing

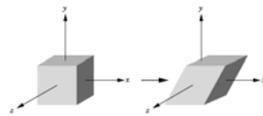
- ❖ Sarebbe derivabile dalle altre...
- ❖ Lo spostamento e' proporzionale alla coord y;

$$x' = x + y \cot \theta$$

$$y' = y$$

$$z' = z$$

$$H_{xy}(\theta) = \begin{bmatrix} 1 & \cot \theta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

23

OpenGL

- ❖ Nello Stato di OpenGL ci sono tre matrici 4x4 di trasformazioni
 - ❖ ModelView
 - ❖ Project
 - ❖ Texture
- ❖ Una di queste e' sempre la matrice corrente
- ❖ Tutte i comandi che modificano matrici fanno sempre riferimento alla matrice corrente

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

24

Opengl

- ❖ Per cambiare matrice Corrente
- ❖ `glMatrixMode()`
 - ❖ `GL_MODELVIEW`
 - ❖ `GL_PROJECTION`
 - ❖ `GL_TEXTURE`
- ❖ Per rimpiazzare la matrice
 - ❖ `glLoadIdentity()`
 - ❖ `glLoadMatrix(Pointer to a matrix);`
- ❖ Tutti gli altri comandi modificano (moltiplicano per un'altra matrice) la matrice corrente.

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

25

OpenGL

- ❖ Leggere una matrice
- ❖ `glGetDoublev(GL_MODELVIEW_MATRIX, pointer_to_matrix)`
- ❖ Nota: nello scambio diretto di matrici OpenGL assume che siano memorizzate in column major order (e.g. per colonne)

$$\begin{bmatrix} a_0 & a_4 & a_8 & a_{12} \\ a_1 & a_5 & a_9 & a_{13} \\ a_2 & a_6 & a_{10} & a_{14} \\ a_3 & a_7 & a_{11} & a_{15} \end{bmatrix}$$

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

26

OpenGL

- ❖ Rotazioni
 - ❖ `glRotatef(angle, ax, ay, az);`
- ❖ Traslazioni
 - ❖ `glTranslatef(dx, dy, dz);`
- ❖ Scalature
 - ❖ `glScalef(sx, sy, sz)`
- ❖ Generica
 - ❖ `glMultMatrix(matrix_pointer)`

9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

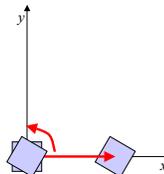
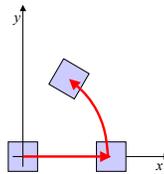
27

Composizione di Matrici in Opengl

- ❖ I comandi opengl postmoltiplicano la matrice corrente per quella specificata

$q = RTp$
`glRotate`
`glTranslate`

$q = TRp$
`glTranslate`
`glRotate`



9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

28

Stack di matrici

- ❖ La matrice corrente puo' essere salvata in uno stack e recuperata successivamente
- ❖ `glPushMatrix()`
- ❖ `glPopMatrix()`

9 Ott 2002

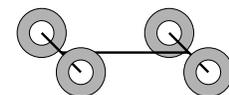
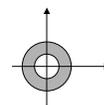
Costruzione di Interfacce - Paolo Cignoni

29

Stack di Matrici

- ❖ Il Matrix Stack e' particolarmente utile quando si disegnano scene strutturate

```
// Whole car transformation
// eg position of the car on the road
glTranslate(CarPos)
glRotate(CarDirection, 0, 1, 0)
For (i=0; i<4; ++i)
{
    glPushMatrix();
    // relative single wheel position
    glTranslate(wheelpos[i]);
    DrawWheel(i);
    glPopMatrix();
}
```



9 Ott 2002

Costruzione di Interfacce - Paolo Cignoni

30