Bài 4B:
Phép biến đổi trong không gian

Ma trận biến đổi 3 chiều
3D Matrix Transformations

- Các phép biến đổi chuyển vị - translation, tỉ lệ-scaling và quay-rotation sử dụng trong không gian 2D đều co thể mở rộng trong không gian 3D
- Again, using homogeneous coordinates it is possible to represent each type of transformation in a matrix form
- In 3D, each transformation is represented by a 4x4 matrix

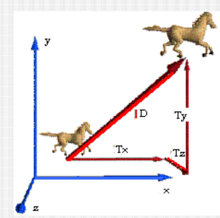
Các phép biến đổi hình học 3 chiều

- **Biểu diễn điểm trong không gian 3 chiều**
  - [x* y* z* h] = [x y z 1]. [T]
  - [x' y' z' 1]= [x*/h y*/h z*/h 1][T]
- **Ma trận biến đổi**

$$[T] = \begin{bmatrix} a & b & c & p \\ d & e & f & q \\ g & i & j & r \\ l & m & n & s \end{bmatrix}$$

Phép tịnh tiến

$$[T(dx,dy,dz)] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ dx & dy & dz & 1 \end{bmatrix}$$



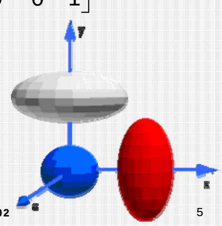
- [X'] = [X] . [T(dx,dy,dz)]
- [x' y' z' 1] =
- [x y z 1].[T(dx,dy,dz)]
- = [x+dx y+dy z+dz 1]

Phép tỉ lệ

$$[x' \quad y' \quad z' \quad 1] = [x \quad y \quad z \quad 1] \begin{bmatrix} s1 & 0 & 0 & 0 \\ 0 & s2 & 0 & 0 \\ 0 & 0 & s3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

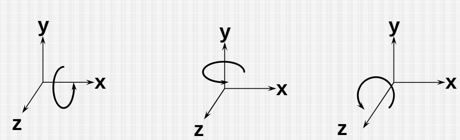
$$= [x.s1 \quad y.s2 \quad z.s3 \quad 1]$$

- s1, s2, s3 là các hệ số tỉ lệ tương ứng trên các trục toạ độ



Rotation

- In 2D, the only rotation possible was about the origin.
- In 3D, there are 3 possible rotations, one about each of the x, y and z axes
- Positive rotations are anti-clockwise, negative rotations are clockwise, when looking down a positive axis towards the origin



Phép quay 3 chiều

- Quay quanh các trục toạ độ
  - Quay quanh trục x

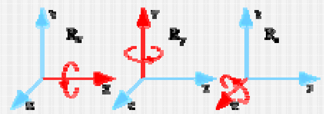
$$[Tx]=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos\phi & \sin\phi & 0\\ 0 & -\sin\phi & \cos\phi & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

  - Quay quanh trục z

$$[Tz]=\begin{bmatrix}\cos\varphi & \sin\varphi & 0 & 0\\ -\sin\varphi & \cos\varphi & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

Quay quanh trục y

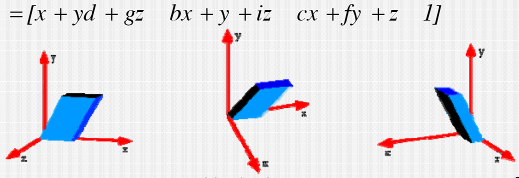
$$[Ty]=\begin{bmatrix}\cos\theta & 0 & -\sin\theta & 0\\ 0 & 1 & 0 & 0\\ \sin\theta & 0 & \cos\theta & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$



Phép biến dạng
(secondary translation)

$$[x' \quad y' \quad z' \quad 1]=[x \quad y \quad z \quad 1]\begin{bmatrix}1 & b & c & 0\\ d & 1 & f & 0\\ g & i & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

$$=[x+yd+gz \quad bx+y+iz \quad cx+fy+z \quad 1]$$

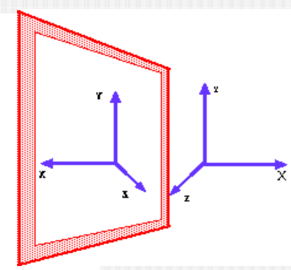


Phép lấy đối xứng
(reflections-secondary translation)

$$M_x=\begin{bmatrix}-1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

$$M_{xy}=\begin{bmatrix}-1 & 0 & 0 & 0\\ 0 & -1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

$$M_{xyz}=\begin{bmatrix}-1 & 0 & 0 & 0\\ 0 & -1 & 0 & 0\\ 0 & 0 & -1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$



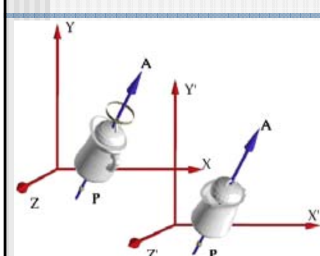
Quay quanh một trục bất kỳ song song với các trục tọa độ

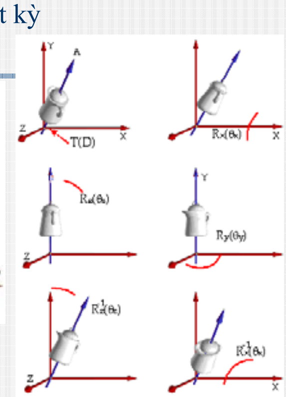
$$[Tr]=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & y & z & 1\end{bmatrix} \Rightarrow [T(\phi)]=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos\phi & \sin\phi & 0\\ 0 & -\sin\phi & \cos\phi & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

$$[Tr]^{-1}=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & -y & -z & 1\end{bmatrix}$$

$$[Th]=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos\phi & \sin\phi & 0\\ 0 & -\sin\phi & \cos\phi & 0\\ 0 & y(1-\cos\phi)+z\sin\phi & z(1-\cos\phi)-y\sin\phi & 1\end{bmatrix}$$

Quay quanh một trục bất kỳ





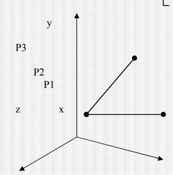
Solution

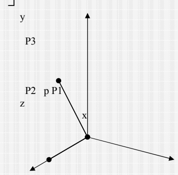
- Chuyển P1 về gốc tọa độ.
- Quay quanh trục y sao cho P1P2 nằm trên mặt phẳng (y, z)
- Quay quanh trục x sao cho P1P2 trùng với trục z.
- Quay quanh trục z sao cho P1P3 nằm trên mặt phẳng (y, z)

Euler’s Theorem: Every rotation around the origin can be decomposed into a rotation around the *x*-axis followed by a rotation around the *y*-axis followed by a rotation around the *z*-axis.

Bước 1: Chuyển P1 về gốc tọa độ

$$[T(-x1,-y1,-z1)]=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ -x1 & -y1 & -z1 & 1\end{bmatrix}$$



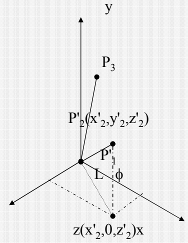


Bước 2: Quay quanh trục y

- cos(- 90 + ϕ) = sinϕ = z'2/L = (z2 - z1)/L
- sin(- 90 + ϕ) = - cosϕ = x'2/L = (x2 - x1)/L

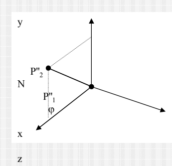
$$L=\sqrt{(z'_2)^2 \quad (x'_2)^2}=\sqrt{(z_2-z_1)^2 \quad (x_2-x_1)^2}$$

- [P"2] = [P'2][T(ϕ-90)]
- = [0 y2-y1 L]



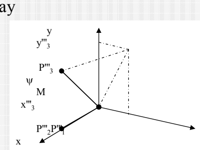
Bước 3: Quay quanh trục x.

- cos φ = z"2/N, sin φ = y"2/N
- Với N = | P"1P"2| là độ dài của đoạn P"1P"2
- $[P'''_2]$ = [P"2][T(φ)] = [P'2][T(ϕ-90)][T(φ)]
- = [P2][T(-x1,-y1,-z1][T(ϕ-90)][T(φ)]
- = [0 0 |P1P2| 1]



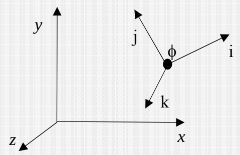
Bước 4: Quay quanh trục z

- $[P'''_3]$= $[P_3]$[T(-x1,-y1,-z1][T(ϕ-90)][T(φ)]
- Với góc quay dương ψ trên trục z
- cos ψ = y_3'''/M; sin ψ = x_3'''/M;
- Ma trận tổng hợp của các phép biến đổi [T] có dạng sau đáp ứng toàn bộ quá trình biến đổi quay đối tượng quanh một trục bất kỳ.
- [T] = [T(-x1,-y1,-z1)][T(ϕ-90)][T(φ)][T(ψ)]



- Kết quả sau biến đổi cần phải đưa về vị trí ban đầu qua các phép biến đổi ngược.
- [Tth]= [T(-x1,-y1,-z1)]x[T(ϕ-0)]x[T(φ)]x [T(ψ)]x[T(ψ)]x[T(φ)]x
- [T(ϕ-90)]x[T(-x1,-y1,-z1)]

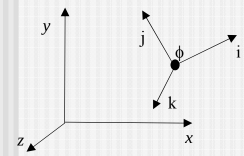
Hệ toạ độ
Coordinate Frame



- Coordinate frame is given by origin ϕ and three mutually orthogonal unit vectors, i, j, k.
- Mutually orthogonal (dot products): i•j = ?; i•k = ?; j•k = ?.
- Unit vectors (dot products): i•i = ?; j•j = ?; k•k = ?.

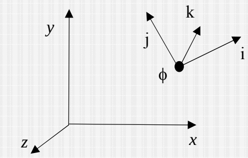
Orientation

Right handed coordinate system:



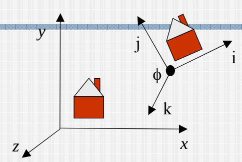
Cross product: i x j = ?

Left handed coordinate system:



Cross product: i x j = ?

Coordinate Transformations



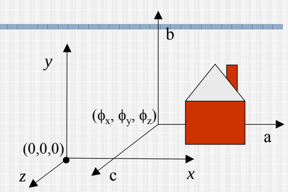
Transform (x, y, z, 0) coordinate frame to (i, j, k, ϕ) coordinate frame.

Affine transformation matrix:

$$\begin{bmatrix} i_x & j_x & k_x & \phi_x \\ i_y & j_y & k_y & \phi_y \\ i_z & j_z & k_z & \phi_z \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Maps (i, j, k, ϕ) coordinates into (x, y, z, 0) coordinates!

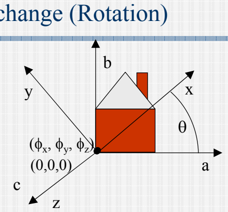
Coordinate change (Translation)



Change from (a,b,c,ϕ) coordinates to (x,y,z,0) coordinates:

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{bmatrix} & & & \\ & & & \\ & & & \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} a \\ b \\ c \\ 1 \end{pmatrix}.$$

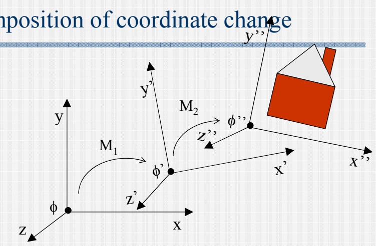
Coordinate change (Rotation)



Change from (a,b,c,ϕ) coordinates to (x,y,z,0) coordinates:

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{bmatrix} & & & \\ & & & \\ & & & \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} a \\ b \\ c \\ 1 \end{pmatrix}.$$

Composition of coordinate change



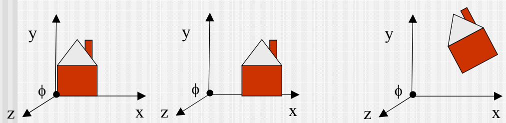
M_1 changes from coordinate frame (x,y,z,θ) to (x',y',z',θ').

M_2 changes from coordinate frame (x',y',z',θ') to (x'',y'',z'',θ'').

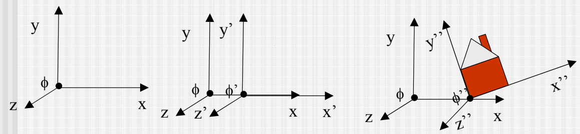
Change from coordinate frame (x,y,z,θ) to (x'',y'',z'',θ''): ?

Object vs. coordinate transformations

Translate and then rotate object:

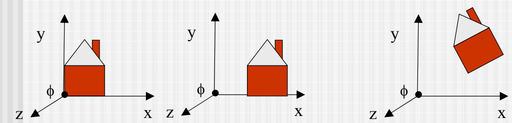


Translate and then rotate coordinate frame:

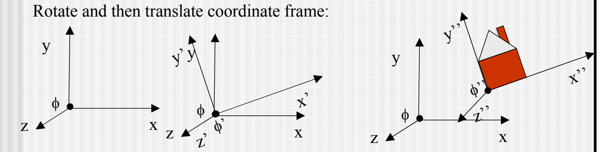


Object vs. coordinate transformations (2)

Translate and then rotate object:



Rotate and then translate coordinate frame:



Order of transformations

Let M_i be the transformation matrix for transformation T_i.

Sequence of object (point) transformations, T_1, T_2, T_3.
Transformation matrix = M_3 x M_2 x M_1.

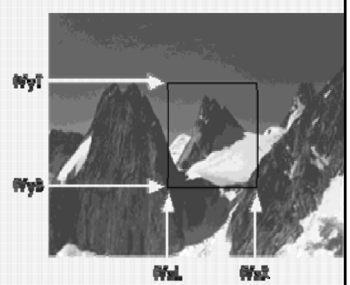
Sequence of coordinate system transformations, T_1, T_2, T_3.
Transformation matrix = M_1 x M_2 x M_3.

Note: OpenGL, OpenInventor use coordinate system transformations.

Hệ tọa độ thực

(WCS-World Coordinate System)

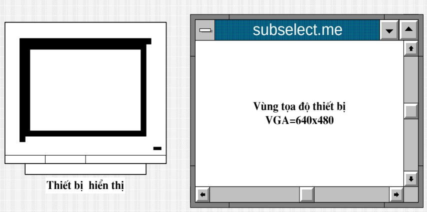
- Là hệ tọa độ của đối tượng được các chương trình ứng dụng sử dụng để mô tả tọa độ của các đối tượng trong thế giới thực.
- Đơn vị trong hệ thống tọa độ phụ thuộc vào không gian và kích thước của đối tượng được mô tả, có thể từ A^0, nm, mm ... đến m, km ...



Hệ tọa độ thiết bị

(DCS-Device Coordinate System)

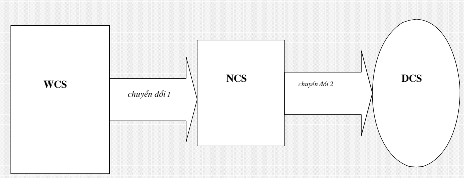
- Là hệ thống tọa độ của thiết bị nơi hiển thị hình ảnh và không gian của đối tượng mà ứng dụng mô tả.



Hệ tọa độ chuẩn

(NCS - Normalized Coordinate System)

- Giải quyết vấn đề khi ứng dụng chạy trên các thiết bị khác nhau
- Có kích thước 1x1



Display Co-ordinates

- The integer, (x, y) screen co-ordinates are far too restrictive to be used to describe physical objects and scenes, because:
  - objects and scenes would need to be remodelled for every new view, or if the device resolution is changed
  - The range of coordinate values (e.g: 640x480, 1024x768 pixels) is inappropriate for many scenes

MODELLING Co-ordinates

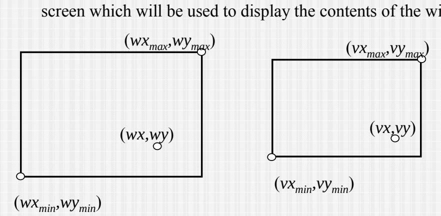
- More logical to use dimensions which are appropriate to the object. e.g.
  - metres for buildings,
  - millimetres for assembly parts,
  - nanometres or microns for molecules, cells, and atoms
- Objects are described with respect to their actual physical size in the real world,
- These measurements are then mapped onto screen co-ordinates before displaying

Basic Viewing Transform

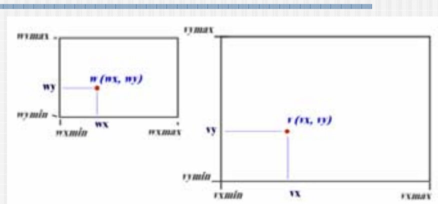
- Apply transform to convert from Modelling co-ordinates to Screen Coordinates
- We can scale dimensions to change the resulting view
- We can even achieve a zooming in and out effect without changing the model by scaling dimensions proportionally
- Vấn đề:
- How much of the model should be drawn?
- Where should it appear on the display?
- How do we convert Real-world co-ordinates into screen co-ordinates?

2-Dimensional Views

- A 2-dimensional view consists of two rectangles:
  - A **Window** given in real world co-ordinates, defining the portion of model (scene) to be drawn
  - A **Viewport** given in screen co-ordinates, defining the portion of the screen which will be used to display the contents of the window



Phép chuyển đổi



$$vx = Vx_{\min} + \frac{(Vx_{\max} - Vx_{\min})(wx - Wx_{\min})}{Wx_{\max} - Wx_{\min}}$$

$$vy = Vy_{\min} + \frac{(Vy_{\max} - Vy_{\min})(wy - Wy_{\min})}{Wy_{\max} - Wy_{\min}}$$

Phép biến đổi theo ma trận

- Ma trận chuyển vị theo Window

$$[T1] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -Xw & -Yw & 1 \end{bmatrix}$$

- Ma trận biến đổi tỉ lệ

$$[S1] = \begin{bmatrix} \frac{\text{Xvmax- Xvmin}}{\text{Xwmax- Xwmin}} & 0 & 0 \\ 0 & \frac{Yv\max - Yv\min}{Yw\max - Yw\min} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

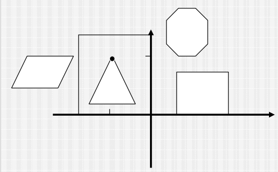
- Ma trận chuyển vị theo tọa độ viewport

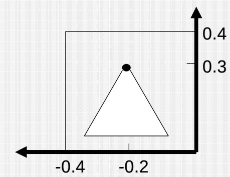
$$[T2] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ Xv & Yy & 1 \end{bmatrix}$$

Ma trận biến đổi tổng hợp của phép chuyển đổi tọa độ

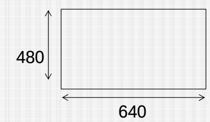
$[T]=[T1]x[S1]x[T2]$

$$[T]=\begin{bmatrix} \frac{Xvmax-Xvmin}{Xwmax-Xwmin} & 0 & 0 \\ 0 & \frac{Yvmax-Yvmin}{Ywmax-Ywmin} & 0 \\ Xvmin-Xwmin\frac{Xvmax-Xvmin}{Xwmax-Xwmin} & Yvmin-Ywmin\frac{Yvmax-Yvmin}{Ywmax-Ywmin} & 1 \end{bmatrix}$$





- Ánh xạ cửa sổ trên tới màn hình 640x480 viewport



Bài tập

Wx=-.2
Wy=.3
Wxmax=0
Wxmin=-0.4
Wymax=0.4
Wymin=0

Vxmax=640
Vxmin=-0.4
Vymax=0
Vymin=480

Vx= (-.2-(-0.4))*(640-0) + 0 = 320
0-(-0.4)
Vy= (.3-0)*(0-480) + 480 = 120
0-(-0.4)