Computer Graphics and Image Processing Geometry III

Part 1 – Lecture 4

Today's Outline

- Homogeneous Coordinates
- 3D Affine Transformations
- Examples

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} \rightarrow \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \qquad \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} \rightarrow \begin{pmatrix} x/w \\ y/w \\ z/w \end{pmatrix}$$

HOMOGENEOUS COORDINATES

Homogeneous Coordinates

- Affine transformation **F** consists of a linear transformation and a translation:
  F(**p**) = **M p** + **t**
- Can we get rid of the vector addition so that **F**(**p**) = **M p** works even for translations?

Wanted: Representation of translations as matrices

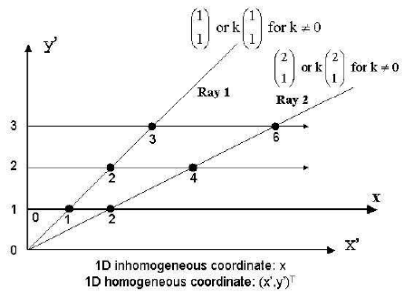
Solution: Homogeneous Coordinates

- Add an additional coordinate to every vector, which is set to 1 if the vector can be translated $\mathbf{p} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$
- Normally we just ignore the additional coordinate
- Also add another row and column to our matrices, e.g. $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

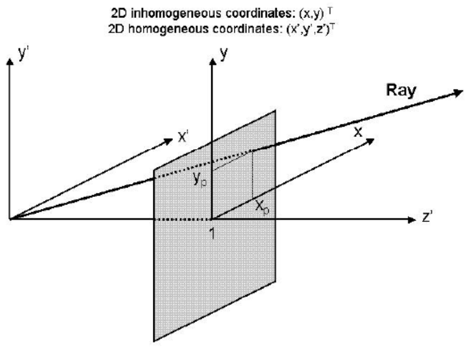
Homogeneous coordinate 1D-2D

In inhomogeneous 1D (cartesian) coordinates, a point is represented by a single value (for example, x=1).

In homogeneous coordinates, a 1D point is represented by a 2D vector $\begin{pmatrix} x' \\ y' \end{pmatrix}$ or $\begin{pmatrix} \frac{x'}{y'} \\ 1 \end{pmatrix}$, which defines a ray:



In inhomogeneous 2D (cartesian) coordinates, a point is represented by a 2D vector $(x, y)^T$. In homogeneous coordinates, it is viewed as a 3D vector $\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix}$ or multiple of the vector $\begin{pmatrix} \frac{x'}{z'} \\ \frac{y'}{z'} \\ 1 \end{pmatrix}$.



Using Homogeneous Coordinates

1. Every vector gets an additional coordinate with value 1 (if translations should affect the vector, otherwise 0)
2. Every matrix gets an additional row and column (0, …, 0, 1)

For transformations other than translations: no difference

$$\begin{pmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} ax+by \\ cx+dy \\ 1 \end{pmatrix}$$

Now we can formulate translations as matrices **T**:

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ 0 \end{pmatrix} = \mathbf{T}\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x+t_x \\ y+t_y \\ 1 \end{pmatrix}$$

Converting Coordinates

Ordinary to Homogeneous Coordinates:

- Just add another coordinate (often called w-coodinate)
- w=1 if translations are possible, w=0 if not

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} \rightarrow \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

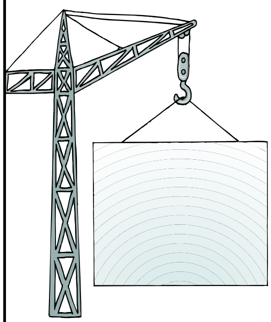
Homogeneous to Ordinary Coordinates:

- Divide all other coordinates by w-coordinate (if $w \neq 0$)
- All homogenous 2D coordinate points $(w\, p_1, w\, p_2, w)^T$ with $w \neq 0$ represent the same ordinary coordinate point $(p_1, p_2)^T$
- Usually w=1, so conversion means just omitting the w-coordinate

$$\begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} \rightarrow \begin{pmatrix} x/w \\ y/w \\ z/w \end{pmatrix}$$

Conversion example

- **Example: the 3D point (5*, 3, 2)^T has the homogeneous representation (5λ, 3 λ, 2 λ, λ)^T with*** arbitrary factor ***λ*** *≠ 0, e.g., (5, 3, 2, 1)^T, or (15, 9, 6, 3)^T, or (−55,−33,−22,−11)^T and so* on.
- Conversely, the homogeneous vector (30*, 10, 15, 5)^T represents the point (6, 2, 3).*
- In homogeneous coordinates projective transformations as well as affine transformations (e.g. translations, rotations, scaling) are specified by linear equations.



3D AFFINE TRANSFORMATIONS

3D Affine Transformations

- Mostly analogous to 2D
- Now we use homogeneous coordinates, i.e. all transformations are represented as a matrix **M** that can be left-multiplied: **M v**

Translation T by a vector **t** = (t_x, t_y, t_z)^T

- Similar to identity matrix
- Rightmost column contains **t**

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Scaling S about origin with scaling factors s_x, s_y, s_z

- Similar to identity matrix
- Diagonal contains scaling factors
- A negative s_x, s_y or s_z causes reflection on the x=0, y=0, or z=0 plane

$$\mathbf{S} = \begin{pmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

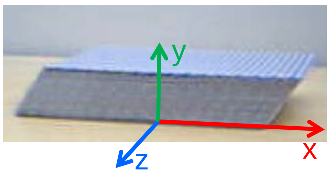
Shearing

General **shearing H**:

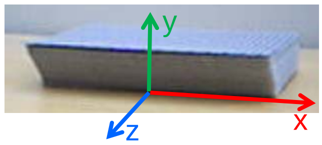
- Any coordinate (x/y/z) can linearly influence any other coordinate
- h_{yx} epresses how much *y* influences *x*

$$\mathbf{H} = \begin{pmatrix} 1 & h_{yx} & h_{zx} & 0 \\ h_{xy} & 1 & h_{zy} & 0 \\ h_{xz} & h_{yz} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Examples:



$h_{yx} > 0$ and all others 0

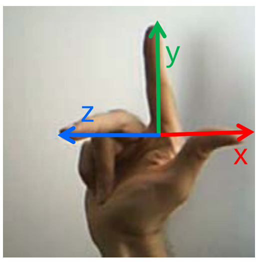


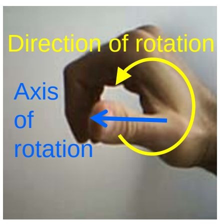
$h_{yz} > 0$ and all others 0

Both together, i.e. $h_{yx} > 0$ and $h_{yz} > 0$ and all others 0

Rotation

- Rotations are the most difficult transformations
- We will consider three different rotation situations:
  1. Rotation around the three coordinate axes (x, y, z)
  2. Rotation to align an object with a new coordinate system
  3. Rotation around an arbitrary axis
- We use a right-handed coordinate system
- We use positive (right-handed) rotation, i.e. counterclockwise when looking into an axis





Rotation around Coordinate Axes

- Three axes (x, y, z) to rotate about, so three different matrices
- Let C = cos θ and S = sin θ, then the matrices for positive (right handed) rotation are:

Rotation about x-axis: $\mathbf{R}_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & C & -S & 0 \\ 0 & S & C & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Rotation about y-axis: $\mathbf{R}_y = \begin{pmatrix} C & 0 & S & 0 \\ 0 & 1 & 0 & 0 \\ -S & 0 & C & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Rotation about z-axis: $\mathbf{R}_z = \begin{pmatrix} C & -S & 0 & 0 \\ S & C & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Note on 3 × 3 rotation matrices:

Row and column corresponding to axis of rotation are as for identity **I**

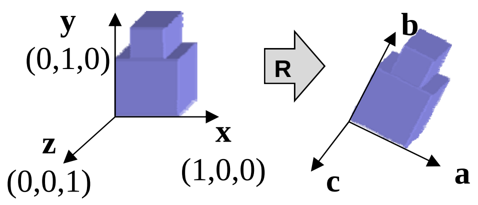
Other elements are C on diagonal, ± S off diagonal, so that **R = I** if θ = 0.

Sign of S can be inferred from the fact that rotation around x,y,z by θ=90^{o} transforms y→z , z→x, x→y, respectively.

Rotating to Align with New Coordinate Axes

Wanted: matrix **R** that rotates the coordinate system to align with a new coordinate system (**a**,**b**,**c**) with the same origin

- **x**,**y**,**z** are the unit vectors of our normal coordinate system
- **a**,**b**,**c** are unit vectors along the axes of the new system



Solution:

R should do the following:

R (1 0 0)^T = **a**

R (0 1 0)^T = **b**

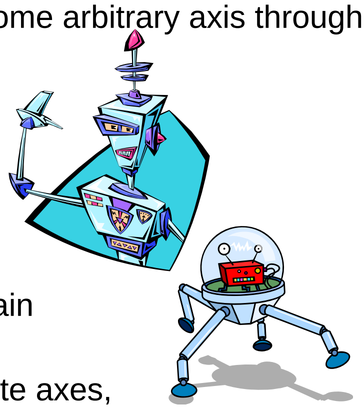
R (0 0 1)^T = **c**

Using homogeneous coordinates:

$$\mathbf{R}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} a_x & b_x & c_x & 0 \\ a_y & b_y & c_y & 0 \\ a_z & b_z & c_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \mathbf{R}$$

Rotation about an Arbitrary Axis

- Often need to rotate an object about some arbitrary axis through a reference point on it
- E.g. forearm of robot rotating around an axis through the elbow
- Involves three steps:
  1. Translate reference point to origin
  2. Do the rotation
  3. Translate reference point back again
- Translation is easy (steps 1 and 3)
- We know how to rotate about coordinate axes, but how about an arbitrary axis through the origin?
  1. Textbook method: decompose rotation into primitive rotations about x,y and z axes
  2. Coordinate system alignment method

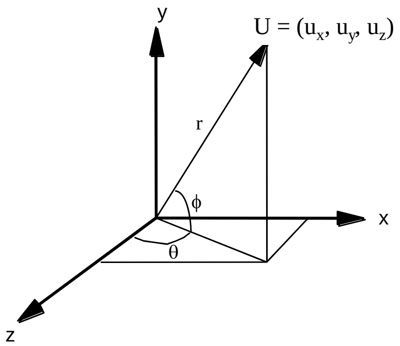


Arbitrary Axis Rotation: Textbook

1. Rotate the object so that the required axis of rotation **r** lies along the z-axis ($\mathbf{R}_{alignZ}$)
2. Do the rotation about z-axis
3. Undo original rotation ($\mathbf{R}_{alignZ}^{-1}$)

How to get $\mathbf{R}_{alignZ}$?

1. Measure azimuth θ as a right handed rotation about the y-axis, starting at the z-axis
2. Measure elevation ϕ (or "latitude") as angle above plane y=0
3. $\mathbf{R}_{alignZ} = \mathbf{R}_x(\phi)\ \mathbf{R}_y(-\theta)$



$$\phi = \tan^{-1}\frac{u_y}{\sqrt{u_x^2 + u_z^2}}$$

$$\theta = atan2(u_x, u_z)$$

(i.e. a four quadrant $\tan^{-1}\frac{u_x}{u_z}$)

Arbitrary Axis Rotation: Alignment

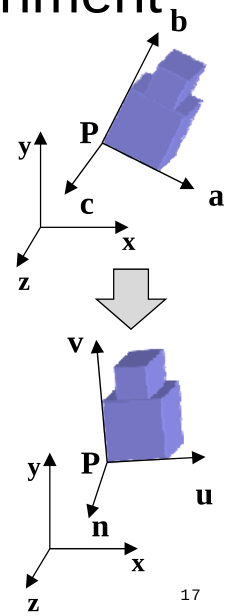
Given:

- Coordinate system **a**,**b**,**c** attached to the object we want to rotate
- Position P of the object's coordinate system
- New coordinate system **u**,**v**,**n** to rotate object to

Solution:

1. Translate object to origin ($\mathbf{T}_P^{-1}$)
2. Rotate **a**,**b**,**c** to align with world coord. axes (inverse of the "rotate to align" case: $\mathbf{R}_{abc}^{-1}$)
3. Rotate coord. axes to align with **u**,**v**,**n** ($\mathbf{R}_{uvn}$)
4. Translate object back to original position ($\mathbf{T}_P$)

Full matrix is: $\mathbf{T}_P\,\mathbf{R}_{uvn}\,\mathbf{R}_{abc}^{-1}\,\mathbf{T}_P^{-1}$



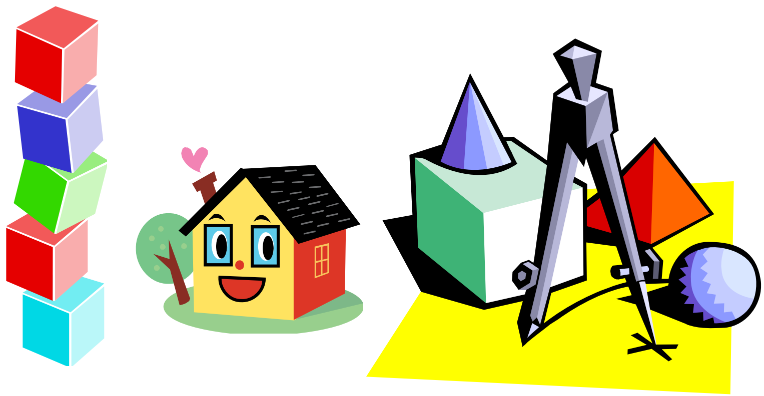
The Inverse of a Rotation Matrix

- Columns of a rotation matrix are unit vectors along the rotated coordinate axis directions
- So columns are orthogonal, i.e. dot products = 0

$$\begin{pmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \end{pmatrix}\begin{pmatrix} a_x & b_x & c_x \\ a_y & b_y & c_y \\ a_z & b_z & c_z \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$\mathbf{R}^T\mathbf{R} = \mathbf{I}$ therefore $\mathbf{R}^{-1} = \mathbf{R}^T$

- The inverse of a rotation matrix is its transpose (matrices with this property are called *orthogonal*)



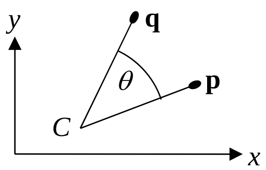
EXAMPLES

Composition of Transformations

- All transformations can be represented as matrix **M**
- Combine several transformations into a single matrix by multiplying all transformation matrixes: $\mathbf{M}_n \mathbf{M}_{n-1} \ldots \mathbf{M}_1 = \mathbf{M}$
- Transformation of rightmost matrix is applied first (i.e. $\mathbf{M}_1$)

Example: rotating an object about its centre point C

1. Translate the object so that its centre is at the origin
2. Rotate about the origin
3. Translate object back to its original position



$$(q_1 \quad q_2 \quad 1)^T = \mathbf{M}(p_1 \quad p_2 \quad 1)^T$$

where

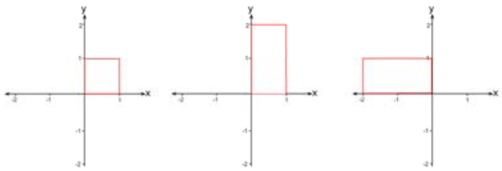
$$\mathbf{M} = \begin{pmatrix} 1 & 0 & c_1 \\ 0 & 1 & c_2 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -c_1 \\ 0 & 1 & -c_2 \\ 0 & 0 & 1 \end{pmatrix}$$

The Order of Transformations Matters!

In general affine transformations do not commute, i.e. **M N** $\neq$ **N M**

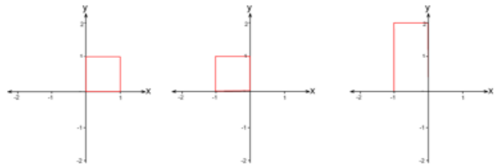
a) First scale by (1,2), then rotate 90º

$$\mathbf{M} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & -2 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$



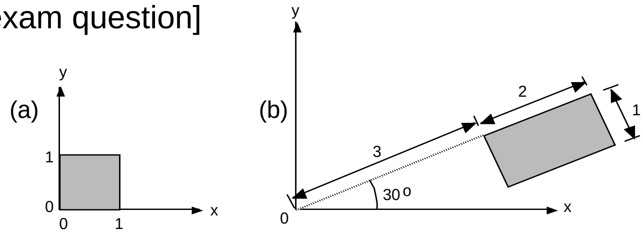
b) First rotate 90º then scale by (1,2)

$$\mathbf{M} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & -1 & 0 \\ 2 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$



Exam Question 1

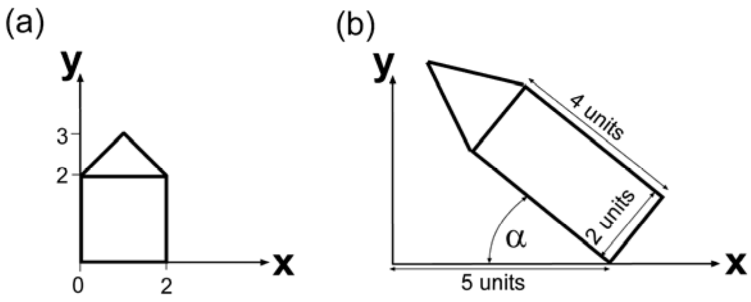
Which homogeneous 2D matrix **M** transforms (a) to (b)?
[1996 exam question]



- Sometimes easier to do these backwards, then take inverse, i.e. starting with figure (b):
  Rotate -30º , shift by (-3,1) , scale by (1/2, 1)
- Hence the required transformation is:
  R(30º) T(3,-1) S(2,1)
  (first scaling, then translation, finally rotation)
- Don't forget to use homogeneous matrices

Exam Question 2

Which homogeneous 2D matrix **M** transforms (a) to (b)?
You are allowed to write M as a product of simpler matrices
(i.e. you need not multiply the matrices) [2003 exam question]



SUMMARY

Summary

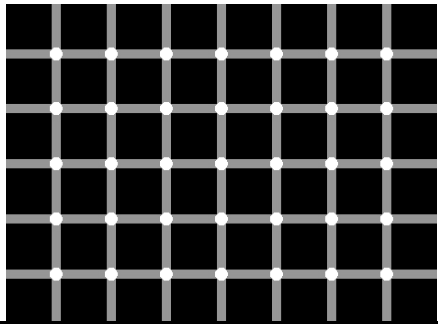
1. **Homogeneous coordinates** make it possible to represent translation as a matrix
2. **3D Affine Transformations** similar to 2D: translation, scaling, shearing and rotation
   - Column vectors of rotation matrix **R** are axis unit vectors of new coord. system to align x,y,z with
   - $\mathbf{R}^{-1} = \mathbf{R}^{T}$
3. Transformations are applied from right to left

References:

- Homogeneous Coordinates: Hill, Chapter 4.5.1
- 3D Affine Transformations: Hill, Chapter 5.3

Quiz

1. An object has a local coordinate system **a**=(1,0,0), **b**=(0,0,-1), **c**=(0,1,0) at position (-10,2,5). Which homogeneous matrix rotates the object into the new coordinate system **u**=(0,-1,0), **v**=(0,0,-1), **n**=(1,0,0)?
2. Solve the sample exam questions 1 and 2.
3. Create your own variation of the exam questions and solve it.



Count the black dots!