M30242 - Graphics and Computer Vision

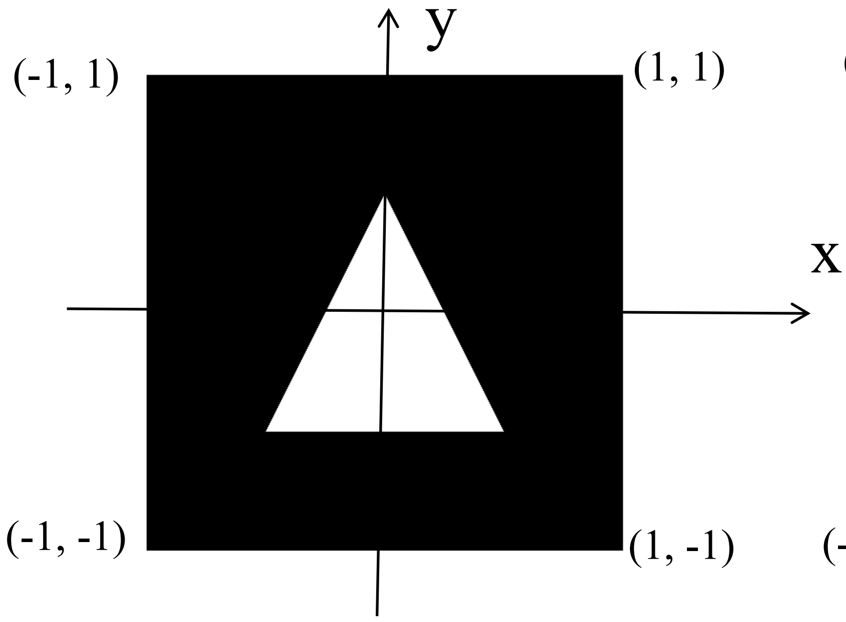
Supplement –1 to Lecture 5:
Coordinates and Transformations

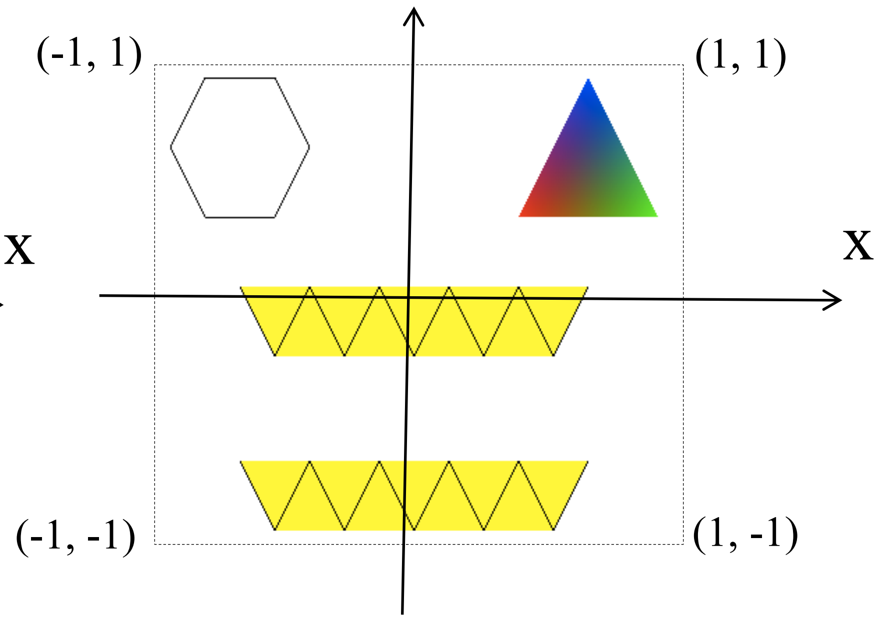
Overview

- Intro: transformations in the graphics pipeline
- Matrix basics
- Points, vectors and homogeneous coorindates
- Translation, rotation and scale
- General transformation

Draw in 2D

- So far, when we draw a shape, we specify the vertex coordinates in a normalised coordinate system.

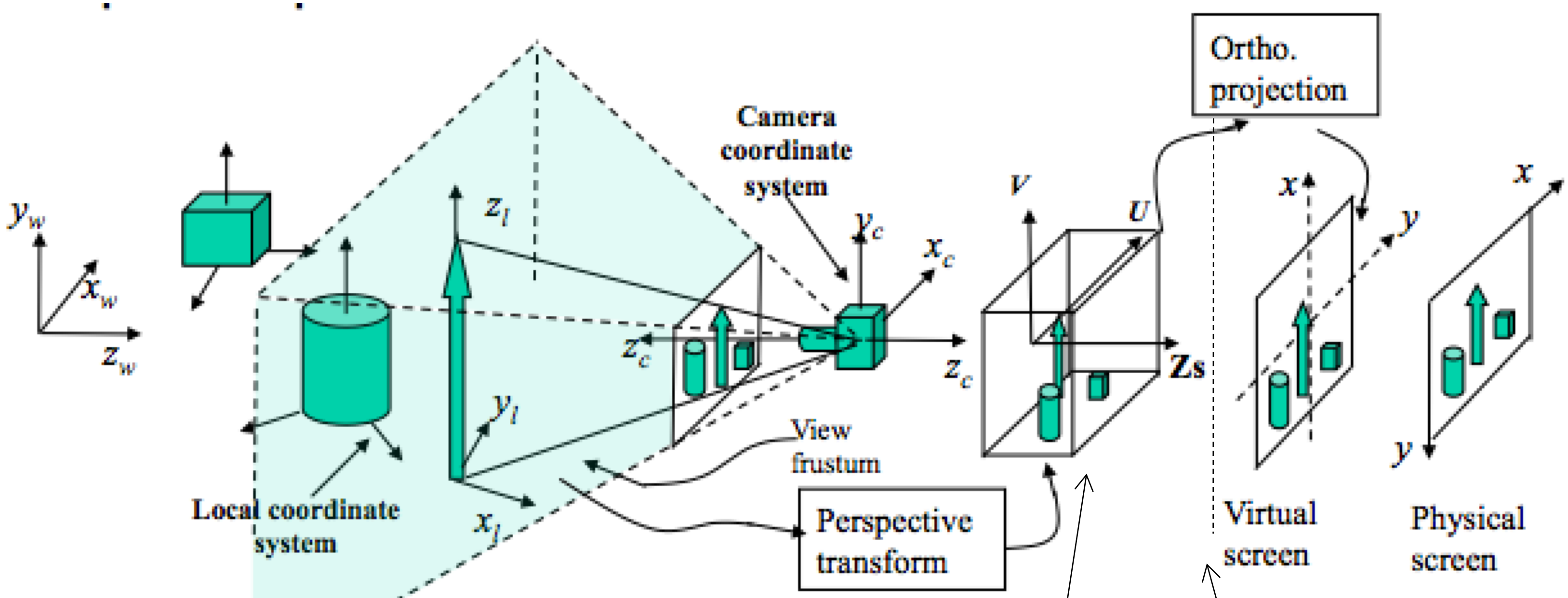




Cont’

- The coordinate system we have used:
  - The range of x, yand z axes is from -1.0 to 1.0. .
- This is the 3D screen coordinate system (ref. graphics pipeline).
- By doing this we have omitted the transformations that should be applied to the vertex coordinates in the vertex shader before they are passed on to the next stage of the graphics pipeline.

Graphics Pipeline & Transformations



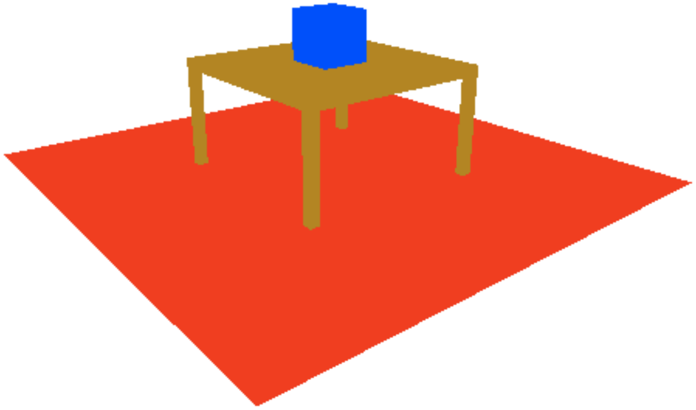
We have defined the shapes in this coordinate system

`gl_Position` receives data at this point

- The clipped coordinates are what the built-in variable `gl_Position` expects after the vertex shader has finished with its job.

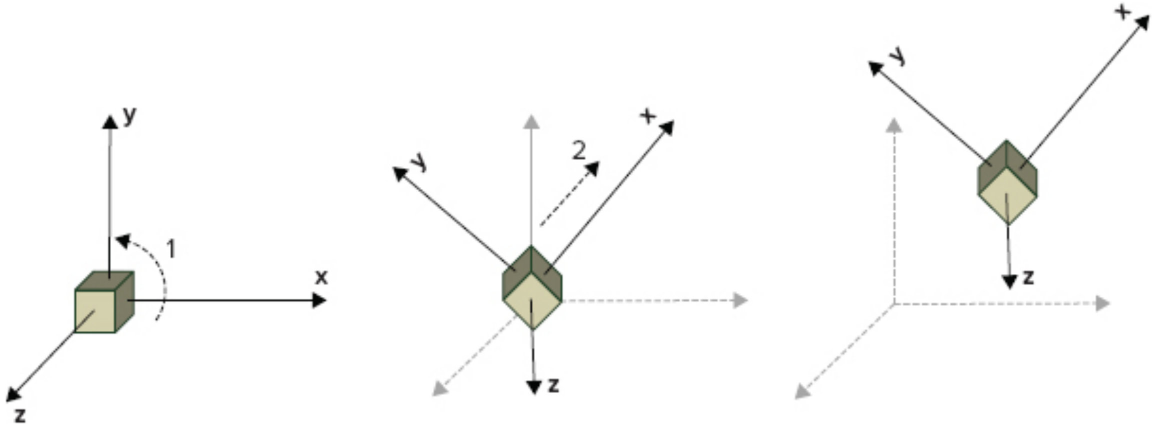
Draw in 3D

- As we go to 3D, is extremely hard to define shapes in the clipped coordinate system, because we need to take the perspective and other effects into account.
- We have to use transformations to control the position, orientation and/or scale of objects and the way by which the objects are drawn (view point & camera parameters).



Transformations

- In a 3D scene objects can be transformed by the following transformations:
  - translation: change its position
  - rotation: change its orientation
  - scale: Change its dimension – make it bigger or smaller, squeezed in one dimension and stretched in another, etc



- These transformations are realised by multiplying coordinates of the vertices (points) of the objects to transformation matrices.

Review: Matrices

- Row matrix (1row X n column) $V = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$

- Column matrix ($n\mathrm{X}1$) $V = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$

  Row & column matrix are also called vectors

- Square matrix ($n\mathrm{X}n$) $M = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$

- General matrix ($m\mathrm{X}n$)

$$M_1 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \qquad M_2 = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

Matrix Multiplication

- Two general matrices:

$$A_{m\times n} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \qquad B_{n\times p} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix}$$

- Their product is a *m-by-p* matrix

$$C_{m\times p} = A_{m\times n} B_{n\times p}$$

whose elements are computed:

$$c_{i,j} = a_{i,1} \times b_{1,j} + a_{i,2} \times b_{2,j} + \cdots + a_{i,n} \times b_{n,j}$$

Matrix Multiplication

- E.g.,

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

$$A \times B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} = \begin{bmatrix} 1\times1+2\times3+3\times5 & 1\times2+2\times4+3\times6 \\ 4\times1+5\times3+6\times5 & 4\times2+5\times4+6\times6 \end{bmatrix} = \begin{bmatrix} 22 & 28 \\ 49 & 64 \end{bmatrix}$$

$$B \times A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} = \begin{bmatrix} 1\times1+2\times4 & 1\times2+2\times5 & 1\times3+2\times6 \\ 3\times1+4\times4 & 3\times2+4\times5 & 3\times3+4\times6 \\ 5\times1+6\times4 & 5\times2+6\times5 & 5\times3+6\times6 \end{bmatrix} = \begin{bmatrix} 9 & 12 & 15 \\ 19 & 26 & 33 \\ 29 & 40 & 51 \end{bmatrix}$$

Cont'd

- Another example:

$$M = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \qquad V = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$M \times V = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \times \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 1 \times 1 + 2 \times 2 + 3 \times 3 \\ 4 \times 1 + 5 \times 2 + 6 \times 3 \\ 7 \times 1 + 8 \times 2 + 9 \times 3 \end{bmatrix} = \begin{bmatrix} 14 \\ 32 \\ 50 \end{bmatrix}$$

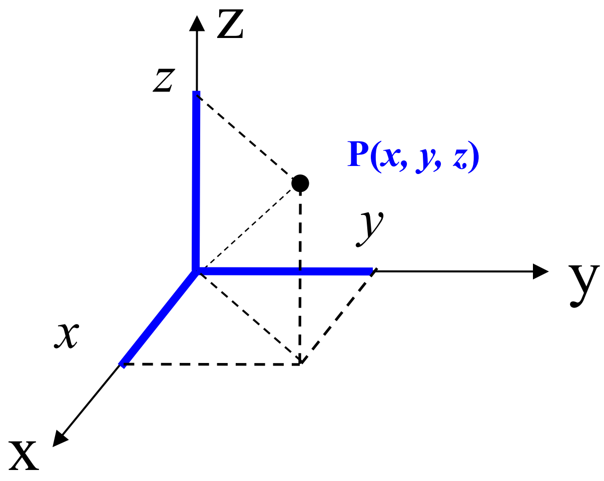
Point Coordinates

- In analytical geometry, a point *P(x,y,z)* is represented by its coordinate values, x, y and z.
- It is written in matrix form (a column matrix):

$$P = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

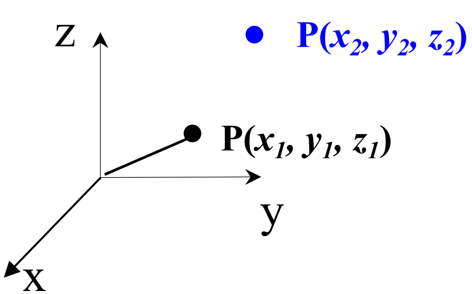
or written as the *transpose* of the column matrix

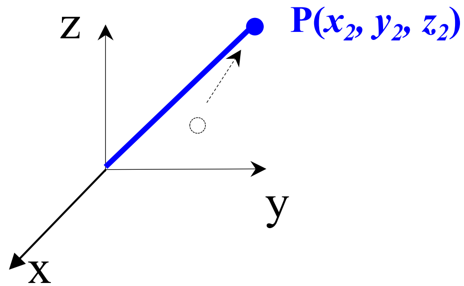
$$P = \begin{bmatrix} x & y & z \end{bmatrix}^T$$



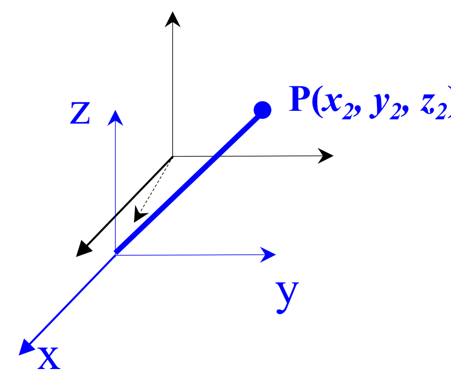
Translate a Point

- In a coordinate system, translating a point means to move it to a new position (This usually means we need to find its coordinates at the new position).
- This change of position can be viewed in two different but equivalent ways:
  - the coordinate system is fixed, and the point is moved
  - the point is fixed, but the coordinate system is moved in opposite direction.





move the point

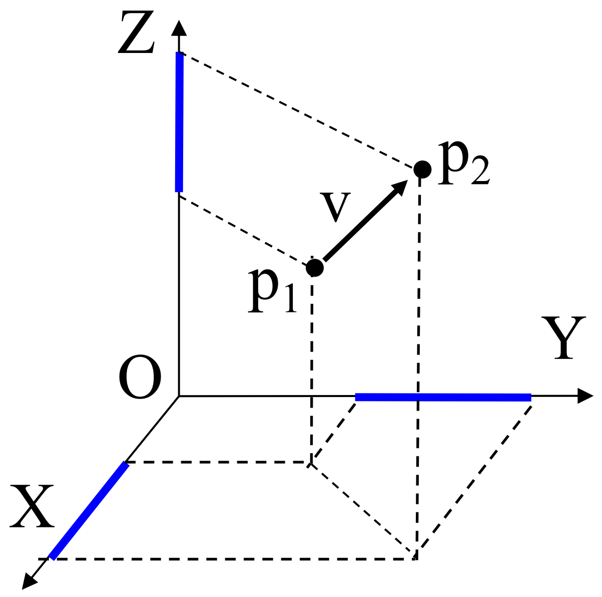


move the coordinate system

Translation Vector

- A translation (the amount and direction of displacement) can be represented as a vector – a geometric entity that has a direction and a length.
- The vector is written in terms of its projections on the coordinate axes (the lengths of blue line segments)

$$V = \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$



Points v.s. Vectors

- By looking at the matrix form of points and translation vector, we cannot really tell which is which. They are all written as column matrices.

$$P = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \qquad V = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$

- A way out of this (and with other more important benefit) is to use homogeneous coordinates

Points in Homogeneous Coord.

- In homogeneous coordinate, a point has a fourth dimension, i.e., its coordinates have the form

$$(x_h, y_h, z_h, w)$$

  where w is a nonzero value.

- The value of w is decided by the following relationship:

$$\left(\frac{x_h}{w}, \frac{y_h}{w}, \frac{z_h}{w}\right) = (x, y, z)$$

  where (x,y,z) is the conventional coordinates of the point.

Cont'd

- For example, a point (2, 3.5, 4) in conventional coordinates can be written in homogeneous coordinates as (4, 7, 8, 2) or (-1, -1.75, -2, -0.5). They are equivalent to (2, 3.5, 4).
- Obviously, a point can have infinite homogeneous coordinates (as long as you can get the original coordinates by dividing the first three coordinates by the 4th coordinate).
- For convenience and uniqueness, the 4th value is set to w=1, i.e., (2, 3.5, 4, 1).

$$P = \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

Vectors in Homogeneous Coord.

- Similar to points, a vector in homogeneous coordinate has a fourth component, but it is *always* set to be *"0"*

$$V = \begin{bmatrix} v_x \\ v_y \\ v_z \\ 0 \end{bmatrix}$$

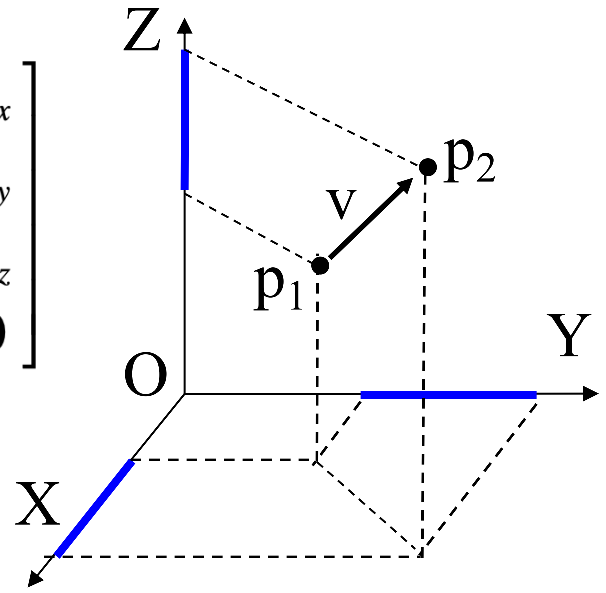
- That means you can tell a point from a translation vector by looking at the 4th coordinate.

Relationship Between Points and Vectors

- Two points in space define a direction and a length – these are the properties of a vector.
- It is obvious, **subtracting a point from another point gets a vector** and **adding a vector to a point gets a new point**

$$V = P_2 - P_1 = \begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} - \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \\ 1 - 1 \end{bmatrix} = \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \\ 0 \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ v_z \\ 0 \end{bmatrix}$$

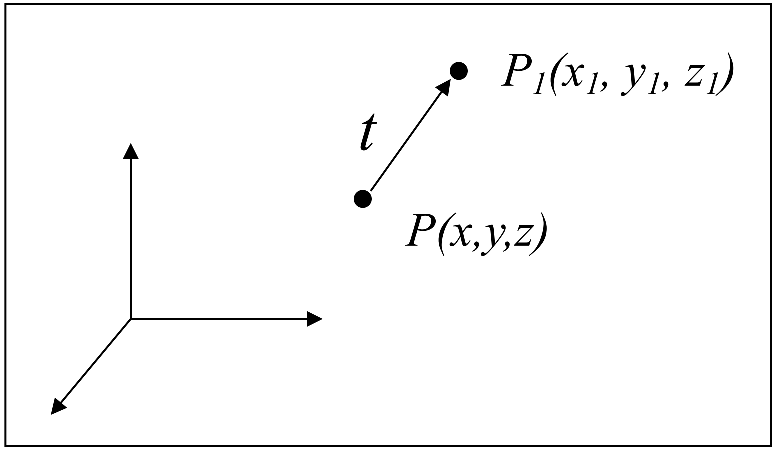
$$P_2 = P_1 + V = \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \\ v_z \\ 0 \end{bmatrix} = \begin{bmatrix} x_1 + x_2 - x_1 \\ y_1 + y_2 - y_1 \\ z_1 + z_2 - z_1 \\ 1 + 0 \end{bmatrix} = \begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix}$$



Translate a Point

- From our discussion, it is clear that translation of a point is matter of adding a vector to the point.
- If a point moves from p to p_1 by a vector t:

$$t = \begin{bmatrix} t_x \\ t_y \\ t_z \\ 0 \end{bmatrix}$$



$$P_1 = \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} + t = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \\ 0 \end{bmatrix} = \begin{bmatrix} x + t_x \\ y + t_y \\ z + t_z \\ 1 \end{bmatrix}$$

Translation Transformation

- Written in matrix form, we have:

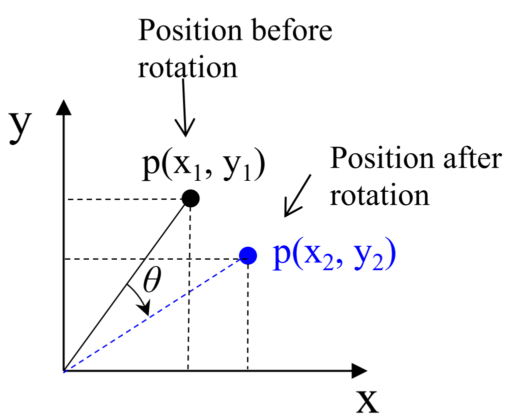
$$P_1 = \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} + t = \begin{bmatrix} x + t_x \\ y + t_y \\ z + t_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

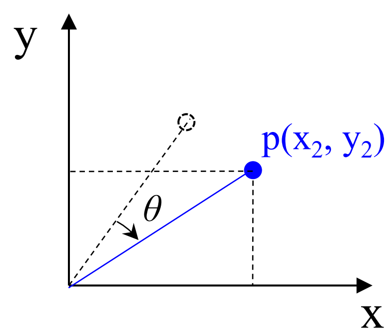
let $T = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$ We call T the translation transformation

- Notice that, if we don't use homogeneous coordinates of points, we cannot write a translation operation as matrix multiplication.
- Translating an object is a matter of multiplying the transformation with the coordinates of all its vertices.

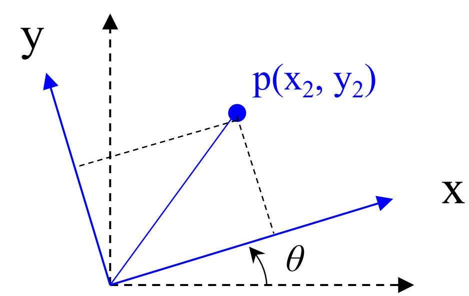
Rotate Around an Axis

- Similar to translating a point, rotating a point around a coordinate axis can also be done in two different but equivalent ways:
  - the coordinate system is fixed and the point is rotated around the the axis perpendicular to the plane of rotation (z axis in this example), or
  - the point is fixed, but the coordinate system is rotated around the axis.





Rotate the point



Rotate the coordinate system

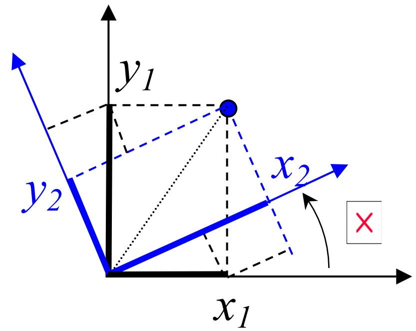
Rotate Around an Axis

- Given a point $p(x_1, y_1, z_1)$ and a coordinate system, the new coordinates of the point, (x_2, y_2, z_2), after rotating the point around an coordinate axis, e.g., z axis, by an angle θ, can be easily computed:

$$x_2 = x_1 \cos\theta + y_1 \sin\theta$$

$$y_2 = -x_1 \sin\theta + y_1 \cos\theta$$

$$z_2 = z_1$$



Writing thre above equations in matrix form, we have:

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}$$

Let $R_z(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Rz(θ) is called rotation transformation around z axis

Cont’d

- Similarly, we can obtain the rotation transformations around x and y axes:

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

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

- These 3x3 matrices are called elementary rotation matrices.

Composite Rotation

- In real application, to achieve the desired orientation, we may need to rotate an object many times.
- Suppose we have a point p(x,y,z), we first rotate it θ_1 about x, then θ_2 about z, then θ_3 about y, then θ_4 about z, etc. We have:

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = R_x(\theta_1) \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

After a rotation of θ_1 about x axis

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = R_z(\theta_2) \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}$$

After a rotation of θ_2 about z axis

$$\begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} = R_y(\theta_3) \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix}$$

After a rotation of θ_3 about y axis

$$\begin{bmatrix} x_4 \\ y_4 \\ z_4 \end{bmatrix} = R_z(\theta_4) \begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix}$$

After a rotation of θ_4 about z axis

Composite Rotation

- We calculate the coordinates of the point following the order of the rotations, we have

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = R_x(\theta_1)\begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad \Longrightarrow \quad \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = R_z(\theta_2)R_x(\theta_1)\begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad \Longrightarrow$$

$$\begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} = R_y(\theta_3)R_z(\theta_2)R_x(\theta_1)\begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad \Longrightarrow \quad \begin{bmatrix} x_4 \\ y_4 \\ z_4 \end{bmatrix} = R_z(\theta_4)R_y(\theta_3)R_z(\theta_2)R_x(\theta_1)\begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

- In general,

$$R(\theta_1,\theta_2,\theta_3,\cdots,\theta_n) = R(\theta_n)\cdots R(\theta_3)R(\theta_2)R(\theta_1) = \begin{bmatrix} m_{00} & m_{01} & m_{02} \\ m_{11} & m_{11} & m_{12} \\ m_{20} & m_{21} & m_{22} \end{bmatrix}$$

  where m_{ij} are some real values.
- The 3x3 matrix, R($\theta_1, \theta_2, \theta_3, \ldots \theta_n$), is called a composite rotation transformation.

Rotation Transformations in Homogeneous Coord.

- If points are given in homogeneous coordinates, we write the matrices in homogeneous form by having one more row at the bottom and one more column at the last column:

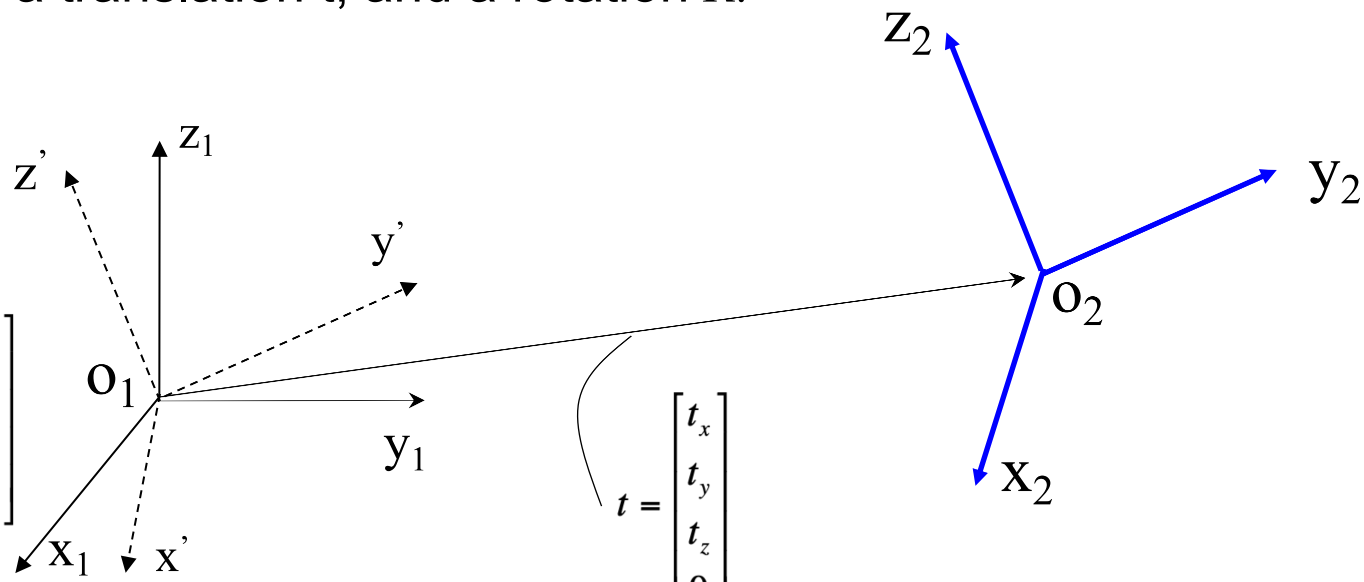
$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad R_z(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- The composite rotation transformation becomes

$$R(\theta_1,\theta_2,\theta_3,\cdots,\theta_3) = \begin{bmatrix} m_{00} & m_{01} & m_{02} & 0 \\ m_{11} & m_{11} & m_{12} & 0 \\ m_{20} & m_{21} & m_{22} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Combining Translation with Rotation

- Suppose we know an object: its position and orientation is represented by its local coordinate system that is at $O_1(x_1, y_1, z_1)$ in terms of the world coordinate system. We now want to move and rotate so that it located and oriented as coordinate system $O_2(x_2, y_2, z_2)$.
- The spatial relationship between the two coordinate systems is defined by a translation t, and a rotation R.



$$R = \begin{bmatrix} m_{00} & m_{01} & m_{02} & 0 \\ m_{11} & m_{11} & m_{12} & 0 \\ m_{20} & m_{21} & m_{22} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$t = \begin{bmatrix} t_x \\ t_y \\ t_z \\ 0 \end{bmatrix}$$

Cont'd

- We obtain the coordinates O_2 (x_2, y_2, z_2) by:
  - first rotating the object by R to obtain an intermediate coordinate system O', which align with O_2,
  - then by translating it by t so that it is at O_2

Step 1:

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = R\begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} m_{00} & m_{01} & m_{02} & 0 \\ m_{10} & m_{11} & m_{12} & 0 \\ m_{20} & m_{21} & m_{22} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} m_{00}x_1 + m_{01}y_1 + m_{02}z_1 \\ m_{10}x_1 + m_{11}y_1 + m_{12}z_1 \\ m_{20}x_1 + m_{21}y_1 + m_{22}z_1 \\ 1 \end{bmatrix}$$

Step 2:

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \\ 0 \end{bmatrix} = \begin{bmatrix} m_{00}x_1 + m_{01}y_1 + m_{02}z_1 + t_x \\ m_{10}x_1 + m_{11}y_1 + m_{12}z_1 + t_y \\ m_{20}x_1 + m_{21}y_1 + m_{22}z_1 + t_z \\ 1 \end{bmatrix}$$

Cont’d

- The computation on the previous slide is equivalent to the following matrix multiplication:

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} m_{00} & m_{01} & m_{02} & t_x \\ m_{11} & m_{11} & m_{12} & t_y \\ m_{20} & m_{21} & m_{22} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} m_{00}x_1 + m_{01}y_1 + m_{02}z_1 + t_x \\ m_{10}x_1 + m_{11}y_1 + m_{12}z_1 + t_y \\ m_{20}x_1 + m_{21}y_1 + m_{22}z_1 + t_z \\ 1 \end{bmatrix}$$

$$\begin{array}{cc} & \begin{array}{cc} \textit{Rotation} & \textit{Translation} \end{array} \\ T = & \begin{bmatrix} m_{00} & m_{01} & m_{02} & t_x \\ m_{11} & m_{11} & m_{12} & t_y \\ m_{20} & m_{21} & m_{22} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

- T combines rotation with translation and is called a ***general transformation***.

Scale

- Scale is a simple operation that changes the dimensions of an object.
- The change is done by multiplying a constant to the coordinates of all vertices. E.g.,
  - multiplying the x-coordinates of the vertices of an object by 2 doubles its size in x-dimension,
  - multiplying the y-coordinates of the vertices of an object by 0.5 halves its size in y-dimension,
- Written in matrix form

$$T = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where s_x, s_y and s_z are scale factors in the x-, y- and z-dimension

Examples

- Double size in x-dimension

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 2x \\ y \\ z \\ 1 \end{bmatrix}$$

- Double size in x-dimension and shrink size in y-dimension by 0.2

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 0.2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 2x \\ 0.2y \\ z \\ 1 \end{bmatrix}$$

- Scale transformation can also be combined with other transformations

$$T = \begin{bmatrix} m_{00}S_x & m_{01} & m_{02} & t_x \\ m_{11} & m_{11}S_y & m_{12} & t_y \\ m_{20} & m_{21} & m_{22}S_z & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Show this by multiplying the general transformation with the general scale transformation.