CS4533 Lecture 3
Slides/Notes

HW1 Discussion; 3D Transformations
(Notes, Ch 2,3,4)

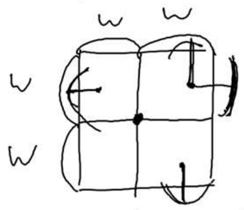
By Prof. Yi-Jen Chiang
CSE Dept., Tandon School of Engineering
New York University

HW1 Part (d)

Input file circles (cx_i, cy_i, r_i) for each circle i. in world coord. system.

Sol. Window width & height are each 2w

Graphics



From input, find

$max\_x = \max_i \{ cx_i + r_i, |cx_i - r_i| \}$

$max\_y = \max_i \{ cy_i + r_i, |cy_i - r_i| \}$

$max\_abs\_coord = \max\{max\_x, max\_y\}$.

*

check for values < 0 to see if the scaling is as desired.

If max_abs_coord > w, then we need to scale down (c_x, c_y, r) for all circles:

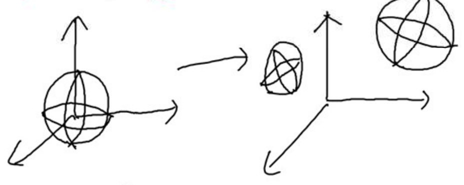
$c_x \leftarrow (c_x \cdot w) / max\_abs\_coord$. $r \leftarrow (r \cdot w)/max\_abs\_coord$.

$c_y \leftarrow (c_y \cdot w) / max\_abs\_coord$.

3D Transformations

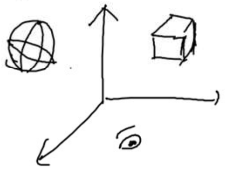
Motivations: Why do we study transformations?

① Modeling Transformation



Scaling
Rotation (change orientation)
Translation (" position)

② Coordinate System change.



World | Frame
coord. system

"what should be the final graphics image when seen from the viewer?"

$$\text{Word Frame} \xrightarrow[\text{transf.}]{} \text{Eye Frame}$$

I. Vector Space, Basis & Linear Transformations.

(1) Basis Vectors

Let V be a vector space.

A set of vectors $\vec{b}_1, \vec{b}_2, \cdots, \vec{b}_n$ $\in V$ is a basis for the vector space V if

① $\vec{b}_1, \vec{b}_2, \cdots \vec{b}_n$ are linearly indept. i.e. $a_1 \vec{b}_1 + a_2 \vec{b}_2 + \cdots + a_n \vec{b}_n = 0$ for scalars $a_1, \cdots a_n$

$\Longleftrightarrow a_1 = a_2 = \cdots = a_n = 0$.

[meaning: any $\vec{b}_i$ can NOT be expressed as a linear combination of the other $n-1$ vectors $\vec{b}_j$, $j \neq i$]

($\vec{b}_1, \vec{b}_2, \cdots \vec{b}_n$ span V)

② Any vector $\vec{v} \in V$ can be expressed as a linear combination of $\vec{b}_1, \cdots, \vec{b}_n$

i.e. $\vec{v} = \sum_{i=1}^{n} c_i \vec{b}_i$ for some scalars $c_1, c_2, \cdots, c_n$

- $\vec{b}_1, \dots \vec{b}_n$ are called the basis vectors.
- The number of basis vectors, n, is called the dimension of the vector space V. In graphics, $n = 3$.
- Consider dimension $n = 3$.

For any vector $\vec{v} \in V$, $\exists$ scalars c_1, c_2, c_3 s.t. $\vec{v} = c_1 \vec{b}_1 + c_2 \vec{b}_2 + c_3 \vec{b}_3$

$$= \underbrace{(\vec{b}_1\ \vec{b}_2\ \vec{b}_3)}_{\text{row matrix}} \underbrace{\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}}_{\text{column matrix}}$$

$\Rightarrow$ $(\vec{b}_1\ \vec{b}_2\ \vec{b}_3)$: row of basis vectors

$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}$: (column) coordinate vector

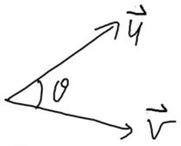
eg. $\vec{v} = (\underset{c_1}{\underline{1}}, \underset{c_2}{\underline{2}}, \underset{c_3}{\underline{3}})$ basis: $\vec{b}_1 = e_1 = (1,0,0)$ X

$\vec{b}_2 = e_2 = (0,1,0)$ Y

$\vec{b}_3 = e_3 = (0,0,1)$ Z

(2) Inner Product & Cross Product.

A. Inner Product. $\vec{u}, \vec{v}$ are vectors



inner product : $\vec{u} \cdot \vec{v}$

$\vec{u} \cdot \vec{v}$ returns a scalar value (real number)

- It allows to define the square length (or length) of a vector $\vec{v}$.

$|\vec{v}|^2 \overset{def}{=} \vec{v} \cdot \vec{v}$ (i.e. $|\vec{v}| = \sqrt{\vec{v} \cdot \vec{v}}$)

- $\vec{u} \cdot \vec{v} = |\vec{u}| \cdot |\vec{v}| \cos\theta$. $\cos\theta = \dfrac{\vec{u} \cdot \vec{v}}{|\vec{u}| \cdot |\vec{v}|}$

Two vector $\vec{u}, \vec{v}$ are orthogonal ($\vec{u} \perp \vec{v}$) if $\theta = 90°$ $\Longleftrightarrow$ $\cos\theta = 0 = \vec{u} \cdot \vec{v}$

- A basis is orthonormal if ① all basis vectors are of length 1. and ② all " " are pairwise orthogonal

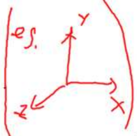
eg.

* In orthonormal basis : $\vec{u} = \sum_i c_i \vec{b}_i$. $\vec{v} = \sum_i d_i \vec{b}_i$ for vectors $\vec{u}, \vec{v}$

where $|\vec{b}_i| = 1 \quad \forall\, i$; $\vec{b}_i \cdot \vec{b}_j = 0 \ \forall\, i \neq j$.

Usually we use such basis

eg.



$\left(\vec{b}_i \cdot \vec{b}_i = |\vec{b}_i|^2 = 1\right)$

$$\vec{u} \cdot \vec{v} = \left(\sum_i c_i \vec{b}_i\right) \cdot \left(\sum_j d_j \vec{b}_j\right) = \sum_i \sum_j (c_i d_j)(\vec{b}_i \cdot \vec{b}_j) = \sum_i \underline{c_i}\, \underline{d_i} \cdot 1$$

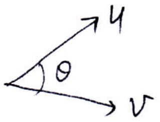
In 3D $\vec{u} = (c_1, c_2, c_3)$

$\vec{v} = (d_1, d_2, d_3)$

$$\underline{\vec{u} \cdot \vec{v} = c_1 d_1 + c_2 d_2 + c_3 d_3}$$

(2) Inner Product & Cross Product

A. Inner Product : u, v are vectors



$\vec{u}\cdot\vec{v}$ returns a scalar value (real number)

* It allows to define the (squared length) (or simply (length)) of a vector:

$|v|^2 := v\cdot v$ (ie $|v| = \sqrt{v\cdot v}$).

* $\vec{u}\cdot\vec{v} = |\vec{u}|\cdot|\vec{v}|\cos\theta$. $\cos\theta = \dfrac{\vec{u}\cdot\vec{v}}{|\vec{u}|\cdot|\vec{v}|}$.

Two vectors $\underbrace{\text{are}}_{\vec{u},\vec{v}}$ (orthogonal) if $\theta = 90°$ $\Leftrightarrow$ $\cos\theta = 0 = \vec{u}\cdot\vec{v}$

* A basis is (orthonormal) if all the basis vectors are $\overset{\text{length} = 1}{\text{(unit length)}}$ and $\overset{\text{of}}{}$ (pairwise orthogonal). Note : Usually we use orthonormal basis

* In orthonormal basis: $\vec{u} = \sum_i c_i\vec{b_i}$ $\vec{v} = \sum_i d_i\vec{b_i}$ where $\vec{b_i}\cdot\vec{b_i} = |\vec{b_i}|^2 = 1$

$\vec{b_i}\cdot\vec{b_j} = 0$ when $i \neq j$

$$\vec{u}\cdot\vec{v} = \left(\sum_i c_i\vec{b_i}\right)\left(\sum_j d_j\vec{b_j}\right) = \sum_i\sum_j (c_i d_j)(\vec{b_i}\cdot\vec{b_j})$$

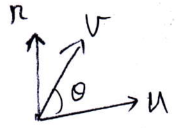
$$= \sum_i c_i d_i$$

In 3D. $\vec{u} = (c_1\ c_2\ c_3)$ $\vec{v} = (d_1\ d_2\ d_3)$ $\vec{u}\cdot\vec{v} = c_1d_1 + c_2d_2 + c_3d_3$

Geometric Meaning:

* ⊙ $\vec{u}\times\vec{v}$ gives a third vector $\perp$ to both $\vec{u}$ and $\vec{v}$

B. Cross Product : In 3D two vectors $\vec{u} = \sum_i c_i\vec{b_i}$ $\vec{v} = \sum_i d_i\vec{b_i}$



$\vec{u}\times\vec{v} = (|\vec{u}|\,|\vec{v}|\sin\theta)\,\vec{n}$. $\vec{n}$ is the (unit vector) perpendicular to the plane spanned by $\vec{u}$ & $\vec{v}$ in the direction by right-hand rule

(4 fingers curling from $\vec{u}$ to $\vec{v}$. the thum points to the direction of $\vec{n}$)

In a (right-handed orthonormal) basis

$$\vec{u}\times\vec{v} = \left(\sum_i c_i\vec{b_i}\right)\times\left(\sum_j d_j\vec{b_j}\right)$$

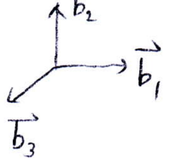
$$= \sum_i\sum_j c_i d_j(\vec{b_i}\times\vec{b_j})$$

$\vec{b_1}\times\vec{b_2} = \vec{b_3}$

$\vec{b_2}\times\vec{b_3} = \vec{b_1}$

$\vec{b_3}\times\vec{b_1} = \vec{b_2}$

$\vec{b_i}\times\vec{b_i} = 0$ $\forall i$



$$\begin{array}{r} (c_1\ \ c_2\ \ c_3) \\ \times)\ (d_1\ \ d_2\ \ d_3) \\ \hline (c_2d_3 - c_3d_2,\ c_3d_1 - c_1d_3,\ c_1d_2 - c_2d_1) \\ \hline \end{array}$$

\#

(3) Linear Transformations

A linear transformation $\mathcal{L}$ is a transformation from V to V satisfying:

① $\mathcal{L}(\vec{u}+\vec{v}) = \mathcal{L}(\vec{u}) + \mathcal{L}(\vec{v})$ ∀ vectors $\vec{u}, \vec{v} \in V$, and

② $\mathcal{L}(a\vec{u}) = a\,\mathcal{L}(\vec{u})$ ∀ vector $\vec{u} \in V$ and real number (scalar) a.

Now let $[\vec{b}_1\ \vec{b}_2\ \vec{b}_3]$ be a basis of the 3D vector space V.

For any $\vec{v} \in V$, we have $\vec{v} = \sum_i c_i \vec{b}_i = [\vec{b}_1\ \vec{b}_2\ \vec{b}_3]\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}$ for some c_1, c_2, c_3

scalar values (real numbers)

$$\mathcal{L}(\vec{v}) = \mathcal{L}\Big(\sum_i c_i \vec{b}_i\Big) = \sum_i c_i\, \mathcal{L}(\vec{b}_i) = [\mathcal{L}(\vec{b}_1)\ \mathcal{L}(\vec{b}_2)\ \mathcal{L}(\vec{b}_3)]\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

But $\mathcal{L}(\vec{b}_1)$ is still a vector in V, thus $\mathcal{L}(\vec{b}_1) = \sum_j M_{j1}\, \vec{b}_j = [\vec{b}_1\ \vec{b}_2\ \vec{b}_3]\begin{bmatrix} M_{11} \\ M_{21} \\ M_{31} \end{bmatrix}$

Similarly for $\mathcal{L}(\vec{b}_2)$ and $\mathcal{L}(\vec{b}_3)$ for some real numbers M_{11}, M_{21}, M_{31}.

$$\therefore\ \mathcal{L}(\vec{v}) = [\mathcal{L}(\vec{b}_1)\ \mathcal{L}(\vec{b}_2)\ \mathcal{L}(\vec{b}_3)]\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = [\vec{b}_1\ \vec{b}_2\ \vec{b}_3]\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

ie Linear transformation

$\vec{v} \Rightarrow \mathcal{L}(\vec{v})$ can be expressed by matrix multiplication:

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} \Rightarrow \underbrace{\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}}_{3\times 3 \text{ matrix } M}\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

(4) Standar Transformations:

Scaling, Rotation, Translation

We like to express each of them by matrix multiplication so that they can be easily supported.

✱ Plan:

① Define the transformation

② Derive $p \longrightarrow q$ using def. and simple geometry

③ Derive matrix M s.t. $q = Mp$.

① Scaling: $p = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$, $q = \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}$ where

$x' = \beta_x x$

$y' = \beta_y y$

$z' = \beta_z z$.

$$\overset{q}{\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}} = \overset{M}{\begin{bmatrix} \beta_x & 0 & 0 \\ 0 & \beta_y & 0 \\ 0 & 0 & \beta_z \end{bmatrix}}\overset{p}{\begin{bmatrix} x \\ y \\ z \end{bmatrix}}$$

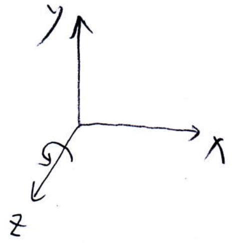
$S = S(\beta_x, \beta_y, \beta_z) = \longrightarrow$ #

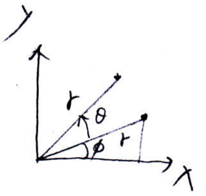
$S^{-1}(\beta_x, \beta_y, \beta_z) = S\left(\frac{1}{\beta_x}, \frac{1}{\beta_y}, \frac{1}{\beta_z}\right)$

② Rotation :

(i) Rotation about the z-axis by an angle of θ

$R_z(\theta)$:





$$\begin{cases} x = r\cos\phi \\ y = r\sin\phi \end{cases} \qquad \begin{cases} x' = r\cos(\phi+\theta) \\ y' = r\sin(\phi+\theta) \\ z' = z \end{cases}$$

$$x' = r\cos(\phi+\theta) = r\left[\cos\phi\cos\theta - \sin\phi\sin\theta\right] = (r\cos\phi)\cos\theta - (r\sin\phi)\sin\theta$$
$$= (\cos\theta)\,x - (\sin\theta)\,y$$

$$y' = r\sin(\phi+\theta) = r\left[\sin\phi\cos\theta + \cos\phi\sin\theta\right] = (r\sin\phi)\cos\theta + (r\cos\phi)\sin\theta$$
$$= (\sin\theta)\,x + (\cos\theta)\,y$$

$$\therefore \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} (\cos\theta)x - (\sin\theta)y \\ (\sin\theta)x + (\cos\theta)y \\ z \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{R_z(\theta)} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \#$$

* Similarly, we can derive $R_x(\theta)$ (rotation about x-axis by an angle θ)
$R_y(\theta)$ (" " y-axis " ")

See next page for details
can skip

* We can undo the rotation by rotating $-\theta$:

$R^{-1}(\theta) = R(-\theta)$ [R = any of R_x, R_y, R_z, and in fact any rotation]

* All $\cos\theta$'s are on the diagonal. $\sin\theta$'s are off diagonal

$\cos(-\theta) = \cos\theta \quad \sin(-\theta) = -\sin\theta \quad \Rightarrow \underline{R^{-1}(\theta) = R(-\theta)} = \underline{R^T(\theta)}$ → transpose

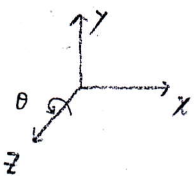
* Any rotation about the origin can be expressed as $R = R_z R_y R_x$

$\underline{R^{-1}} = (R_z R_y R_x)^{-1} = R_x^{-1} R_y^{-1} R_z^{-1} = R_x^T R_y^T R_z^T = (R_z R_y R_x)^T = \underline{R^T}$

$\therefore$ For any rotation matrix $\underline{R^{-1} = R^T}$

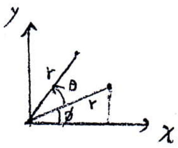
* Not commutative : $R_x R_y \neq R_y R_x$ (use textbook as an example)

Derivation of $R_x(\theta)$, $R_y(\theta)$, $R_z(\theta)$:



Supplementary skip !

(1) $R_z(\theta)$:



$$\begin{cases} x = r\cos\phi \\ y = r\sin\phi \end{cases}$$

$$\begin{cases} x' = r\cos(\phi+\theta) \\ y' = r\sin(\phi+\theta) \\ z' = z \end{cases}$$

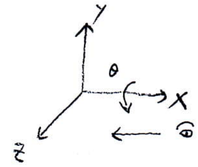
$$x' = r[\cos\phi\cos\theta - \sin\phi\sin\theta]$$
$$= (r\cos\phi)\cos\theta - (\sin\theta)(r\sin\phi)$$
$$= (\cos\theta)x - (\sin\theta)y$$

$$y' = r[\sin\phi\cos\theta + \cos\phi\sin\theta]$$
$$= (\cos\theta)y + (\sin\theta)x$$
$$= (\sin\theta)x + (\cos\theta)y$$

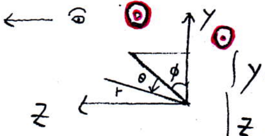
$$\therefore R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

\#

$^{(2)}$ $R_x(\theta)$:



(From x axis at infinity look at y-z plane.



$$\begin{cases} y = r\cos\phi \\ z = r\sin\phi \end{cases}$$

$$y' = r\cos(\phi+\theta) = r[\cos\phi\cos\theta - \sin\phi\sin\theta]$$
$$= (r\cos\phi)\cos\theta - (\sin\theta)(r\sin\phi)$$

same as before

$$= (\cos\theta)y - (\sin\theta)z$$

$$z' = r\sin(\phi+\theta) = r[\sin\phi\cos\theta + \cos\phi\sin\theta]$$
$$= (r\sin\phi)(\cos\theta) + (r\cos\phi)\sin\theta$$

same as before

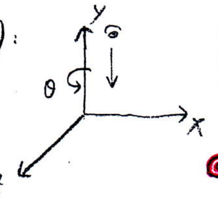
$$= (\sin\theta)y + (\cos\theta)z$$

$$x' = x$$

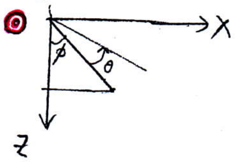
$$\therefore R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

\#

$^{(3)}$ $R_y(\theta)$:



(From y axis at infinity look at the x-z plane:



$$\begin{cases} x = r\sin\phi \\ z = r\cos\phi \end{cases}$$

$$x' = r\sin(\phi+\theta) = r[\sin\phi\cos\theta + \cos\phi\sin\theta]$$
$$= (r\sin\phi)\cos\theta + (r\cos\phi)\sin\theta$$

same as before

$$= (\cos\theta)x + (\sin\theta)z$$

$$z' = r\cos(\phi+\theta) = r[\cos\phi\cos\theta - \sin\phi\sin\theta]$$
$$= (r\cos\phi)\cos\theta - (\sin\theta)(r\sin\phi)$$

same as before.

$$= (-\sin\theta)x + (\cos\theta)z$$

$$y' = y$$

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

\#

③ Translation : $p \xrightarrow{d=(d_x,\, d_y,\, d_z)} q$ $\quad p=\begin{bmatrix}x\\y\\z\end{bmatrix} \quad q=p+d=\begin{bmatrix}x+d_x\\y+d_y\\z+d_z\end{bmatrix}=\begin{bmatrix}x'\\y'\\z'\end{bmatrix}$

Matrix representation : $q=Mp$: $\begin{bmatrix}x+d_x\\y+d_y\\z+d_z\end{bmatrix}=\begin{bmatrix}a&b&c\\d&e&f\\g&h&i\end{bmatrix}\begin{bmatrix}x\\y\\z\end{bmatrix} \Rightarrow x+d_x=ax+by+cz$

$a=1 \quad b=c=0$

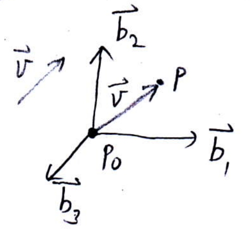
How about d_x ?

⇒ There is NOT enough expressive power to make it work

We need new concepts : Affine space & Homogeneous coordinate System

II. Affine Space and Homogeneous Coordinate System

* Affine Space : vector space + points



Frame (or Coordinate System) : Basis vectors + origin point P_0 $\quad \{\vec b_1, \vec b_2, \vec b_3\}$: basis vectors

* Vectors are NOT fixed w.r.t origin. But each point p is fixed relative to the origin P_0

$P=P_0+\vec v \quad (\vec v = P-P_0)$

Vector $\vec v$ is uniquely expressed by : $\vec v = x\vec b_1 + y\vec b_2 + z\vec b_3 \ (+0\cdot P_0)$

Point P " : $P = P_0 + \vec v = x\vec b_1 + y\vec b_2 + z\vec b_3 + 1\cdot P_0$

$\vec v=(x,y,z)$ the same ? (origin point P_0 is NOT shown)

$P=(x,y,z)$

Sol : Use Homogeneous Coord. System : $P=[\vec b_1\ \vec b_2\ \vec b_3\ P_0]\begin{bmatrix}x\\y\\z\\\textcircled{1}\end{bmatrix} \quad \vec v=[\vec b_1\ \vec b_2\ \vec b_3\ P_0]\begin{bmatrix}x\\y\\z\\\textcircled{0}\end{bmatrix}$

ie Extending to 4D 4th component : $\begin{cases}0 & \text{if vector}\\ 1 & \text{if point}\end{cases}$

Consistency check :	vector + vector = vector	vector + pt = pt	pt + pt = ? (Never happens!)	pt − pt = vector
4th component :	0 + 0 = 0 ✓	0 + 1 = 1 ✓	1 + 1 = 2	1 − 1 = 0 ✓

* Translation : $P=\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$

$$q=\begin{bmatrix}x'\\y'\\z'\\1\end{bmatrix}=\begin{bmatrix}x+d_x\\y+d_y\\z+d_z\\1\end{bmatrix}=\underbrace{\begin{bmatrix}1&0&0&d_x\\0&1&0&d_y\\0&0&1&d_z\\0&0&0&1\end{bmatrix}}_{=T}\overset{P}{\begin{bmatrix}x\\y\\z\\1\end{bmatrix}}=\underbrace{\begin{bmatrix}\begin{bmatrix}\overset{3\times3}{I}\end{bmatrix}&\begin{matrix}d_x\\d_y\\d_z\end{matrix}\\ \begin{matrix}0&0&0\end{matrix}&1\end{bmatrix}}_{=T}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$$

(I : identity matrix)

* $T^{-1}(d_x, d_y, d_z) = T(-d_x, -d_y, -d_z)$

Note : * Using Homogeneous Coord. System and 4×4 matrix multiplication to perform a 3D transformation is called an affine transformation

* Translation is NOT a linear transformation, but is an affine transformation

* We can use affine transformation to perform linear transformation (eg. scaling, rotation) on points:

Let l be a 3×3 matrix for linear transformation (such as scaling, rotation)

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \overset{3\times3,\ l}{\begin{bmatrix} \equiv \end{bmatrix}} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \Rightarrow \overset{q}{\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix}} = \begin{bmatrix} \overset{3\times3,\ l}{\begin{bmatrix} \equiv \end{bmatrix}} & \begin{matrix} 0 \\ 0 \\ 0 \end{matrix} \\ 0\ 0\ 0 & 1 \end{bmatrix} \overset{p}{\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}}$$

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

* Scaling
$S(\beta_x, \beta_y, \beta_z)$:
Fixed pt is at the (origin.)

* Rotation
$R(\theta, \vec{v})$:
rotation about axis (whose vector is $\vec{v}$) going thru origin:
Fixed pt is the (origin)

∴ The corresponding 4×4 matrix L is

$$L = \begin{bmatrix} l & 0 \\ 0 & 1 \end{bmatrix}$$ ✱

⊙ * If l is rotation (resp. scaling), the properties of rotation (resp. scaling) stay the same for L.

cf: The 4×4 matrix T for translation is:

$$T = \begin{bmatrix} \overset{3\times3,\ i}{\begin{bmatrix} \equiv \end{bmatrix}} & \overset{t}{\begin{matrix} d_x \\ d_y \\ d_z \end{matrix}} \\ 0\ 0\ 0 & 1 \end{bmatrix} = \begin{bmatrix} i & t \\ 0 & 1 \end{bmatrix}$$ ✱

where i is a 3×3 identity matrix

t is a column vector $\begin{bmatrix} d_x \\ d_y \\ d_z \end{bmatrix}$

* In general, a 4×4 matrix M for affine transformation is

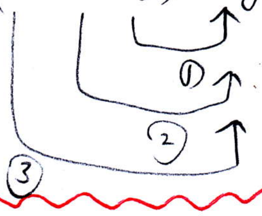
$$M = \begin{bmatrix} \overset{3\times3,\ l}{\begin{bmatrix} \equiv \end{bmatrix}} & \begin{matrix} d_x \\ d_y \\ d_z \end{matrix} \\ 0\ 0\ 0 & 1 \end{bmatrix} = \begin{bmatrix} l & t \\ 0 & 1 \end{bmatrix} = \overset{i}{\begin{bmatrix} \begin{bmatrix} \equiv \end{bmatrix} & \begin{bmatrix} t \end{bmatrix} \\ 0\ 0\ 0 & 1 \end{bmatrix}} \overset{l}{\begin{bmatrix} \begin{bmatrix} \| \| \| \end{bmatrix} & \begin{matrix} 0 \\ 0 \\ 0 \end{matrix} \\ 0\ 0\ 0 & 1 \end{bmatrix}} = T \cdot L \quad (\neq LT)$$

i.e. Decomposed into ① L (scaling $\overset{\text{eg.}}{}$ and/or rotation with fixed pt at the origin), then

② T (translation) ✱

* Concatenation of Transformations

eg. $M \leftarrow M_1 M_2 M_3$. Applying M to obj: ⊙ * $M \cdot obj = (M_1 M_2 M_3) \cdot obj$



Order is very important, since $M_1 M_2 \neq M_2 M_1$

* standard Transformation functions:

glRotatef(angle, V_x, V_y, V_z) ⟷ Rotate(angle, V_x, V_y, V_z)

→ in degrees

○

(The rotation axis goes thru the origin
∴ Origin is the fixed pt.

(specifies the vector of the rotation axis

glScalef(β_x, β_y, β_z) ⟷ Scale(β_x, β_y, β_z)

Fixed pt is at the origin

glTranslatef(d_x, d_y, d_z) ⟷ Translate(d_x, d_y, d_z)

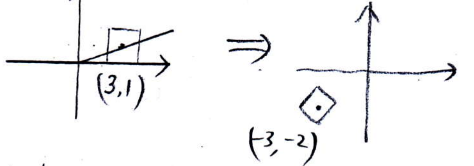
○

* Helper functions
Rotate(), Scale(), Translate()
each returns a 4×4 matrix

* They can be multiplied together

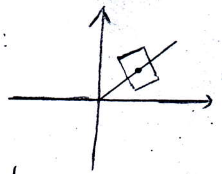
* In general, ⊙ (first translate so that the object center is at the origin,
perform rotations / scalings (with center at the origin)
then translate the obj center to the final location.

Eg 1.



Correct:
① T(-3,-1)
② R(45°)
③ T(-3,-2)

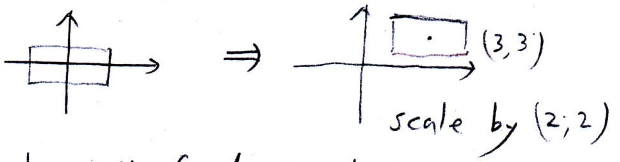
Common mistake:
① R(45°)



center also moves

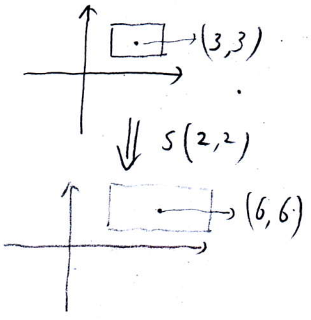
R: rotation about the origin (fixed at origin)

Eg 2



Correct:
① S(2,2)
② T(3,3)

Wrong:
① T(3,3)
② S(2,2)



center also moves

S: scaling with fixed pt at the origin

○ Note: Textbook Sec. 3.6 (Transformation for Normal Vectors)
is skipped here.
Cover it later when discussing shading.