# LABORATORIO DI REALTÀ AUMENTATA

Claudio Piciarelli

Università degli Studi di Udine
Corso di Laurea in Scienze e Tecnologie Multimediali

# The mathematics of 3D graphics

# Linear algebra basic concepts

- **Scalar**: any real number. E.g. 5, -4, 2.14 …

  numeri REALI

- **Vector**: a "row" (or a "column") of scalars.

$$[1,\ 5,\ 7] \qquad \begin{bmatrix} 3 \\ -1 \end{bmatrix} \qquad [-10,\ 0,\ 0,\ 0,\ 3]$$

dim = 3 dim = 2

*three vectors of size 3, 2 and 5 respectively*

Un punto nello spazio 3d è identificato dal vettore [x, y, z]

# Linear algebra basic concepts

- **Matrix:** a "grid" of scalars

$$\begin{bmatrix} 5 & -3 \\ 1 & 2 \end{bmatrix} \quad \begin{bmatrix} 1 & -1 \\ -1 & 4 \\ 2 & 1 \end{bmatrix}$$

*two matrices of sizes 2x2 and 3x2 respectively*

- Size of a matrix: **rows** x **columns**

Vectors can be seen as special matrices where one of the two dimensions is 1

# Addition

- Adding two vectors is done element-by-element. The two vectors must have the same size!
- [ a, b, c ] + [ d, e, f ] = [ a+d, b+e, c+f ]

$$[3, 2] + [1, 1] = [4, 3]$$

$$\begin{bmatrix} -1 \\ 2 \\ 4 \end{bmatrix} + \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 5 \end{bmatrix}$$

# Addition

- Matrix addition has no surprises. The two matrices must have the same size, addition is done element-by-element

$$\begin{bmatrix} 1 & 0 \\ 1 & 2 \end{bmatrix} + \begin{bmatrix} 2 & -2 \\ 3 & 1 \end{bmatrix} = \begin{bmatrix} 3 & -2 \\ 4 & 3 \end{bmatrix}$$

# Multiplication by a scalar

- Vectors and matrices can be multiplied element-by-element by a scalar

$$5\begin{bmatrix}1\\2\\3\end{bmatrix}=\begin{bmatrix}5\\10\\15\end{bmatrix}$$

$$2\begin{bmatrix}-1 & 3\\4 & 2\end{bmatrix}=\begin{bmatrix}-2 & 6\\8 & 4\end{bmatrix}$$

# Dot product

- The dot product (or scalar product) is an operation between two vectors of the same size which returns a scalar

$$\begin{bmatrix} a & b & c \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = ax + by + cz$$

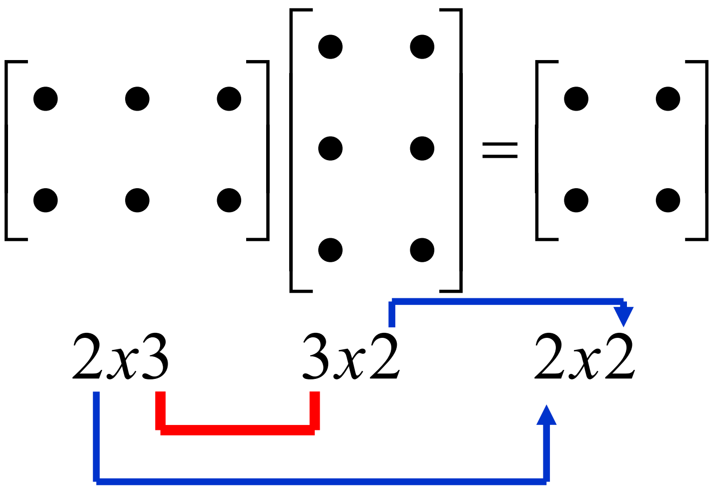
(we will soon understand why we wrote the first as a row vector and the second as a column vector)

# Dot product

- $[1 \quad 2] \cdot \begin{bmatrix} 3 \\ 4 \end{bmatrix} = 1 * 3 + 2 * 4 = 3 + 8 = 11$
- $[-1 \quad 2 \quad 1] \cdot \begin{bmatrix} 3 \\ 1 \\ -2 \end{bmatrix} = -3 + 2 - 2 = -3$
- $[-6 \quad 12 \quad 0.76] \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = 0$

# Matrix multiplication

- If A is an $n \times m$ matrix and B is an $m \times k$ matrix, then A and B can be multiplied, and the result is a $n \times k$ matrix




**The number of columns of the first matrix must be equal to the number of rows of the second matrix!**

# Matrix multiplication

- "rows by columns" method
- The element at row *i* and column *j* in the output matrix is the dot product of the *i*-th row of the first matrix and the *j*-th column of the second matrix

$$\begin{bmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{bmatrix} \begin{bmatrix} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{bmatrix} = \begin{bmatrix} \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet \end{bmatrix}$$

$2x3$ $\quad$ $3x4$ $\quad$ $2x4$

# Matrix multiplication

- Multiplication of a matrix and a vector is just a special case of matrix multiplication
- The same for the dot product of two vectors

# Matrix multiplication

$$\begin{bmatrix} 1 & 2 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} -1 & 1 \\ 2 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} 1 * (-1) + 2 * 2 & 1 * 1 + 2 * 1 \\ 0 * (-1) + 3 * 2 & 0 * 1 + 3 * 1 \end{bmatrix} =$$

$$= \begin{bmatrix} 3 & 3 \\ 6 & 3 \end{bmatrix}$$

# Matrix multiplication

- $\begin{bmatrix} 1 & -1 \\ 0 & 3 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} 3 \\ 5 \end{bmatrix} = \begin{bmatrix} 3-5 \\ 0+15 \\ 6+20 \end{bmatrix} = \begin{bmatrix} -2 \\ 15 \\ 26 \end{bmatrix}$

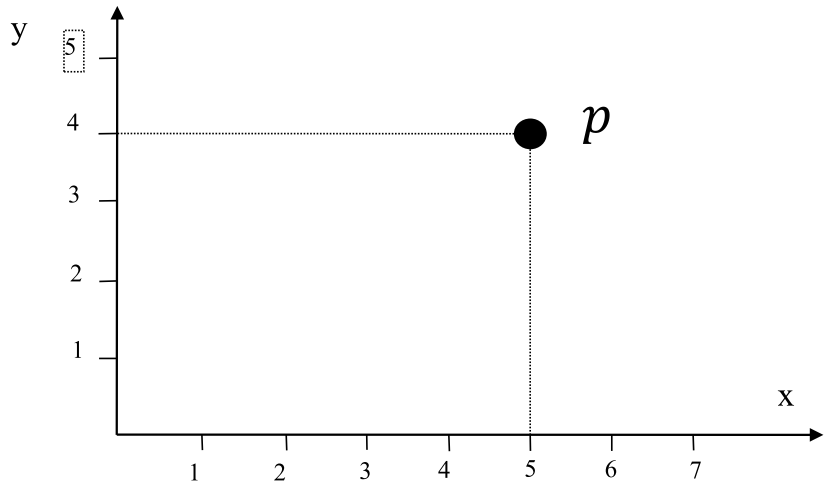
- $\begin{bmatrix} -1 & 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 3 \\ 1 \\ -2 \end{bmatrix} = -3$

# Matrix mult is not commutative

- **WARNING**: matrix multiplication is not commutative
- If A and B are matrices, $AB \neq BA$
- Try with $A = \begin{bmatrix} 1 & 2 \\ 0 & 3 \end{bmatrix}$, $B = \begin{bmatrix} -1 & 1 \\ 2 & 1 \end{bmatrix}$
- Try with $A = \begin{bmatrix} 1 & -1 \\ 0 & 3 \\ 2 & 4 \end{bmatrix}, B = \begin{bmatrix} 3 & 1 \\ 5 & 1 \end{bmatrix}$

# Linear algebra and 3D graphics?

- We are used to denote the position of a point in a 2D space with its coordinates




- A point in a 2D space is a vector of size 2

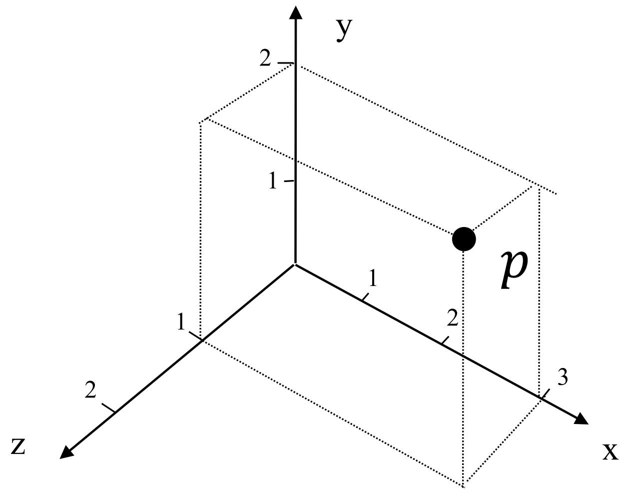
VETTORE

- $p = [\,5, 4]$

$y$

$x$

# Linear algebra and 3D graphics?




- A point in a 3D space is a vector of size 3
- $p = [3, 2, 1]$ ⇒ UN VETTORE RAPPRESENTA UN VERTICE !!

$x$ $y$ $z$

# Linear algebra and 3D graphics?

- Transforming a 3D shape means transforming its vertices → modifico solo i vertici

- The basic transformations (translate, rotate, scale) can be expressed as ***matrix-vector multiplications***

  ↳ vanno per forza espresse come prodotto matrice vettore

# Translate

=> TRASLAZIONE è somma di vettori

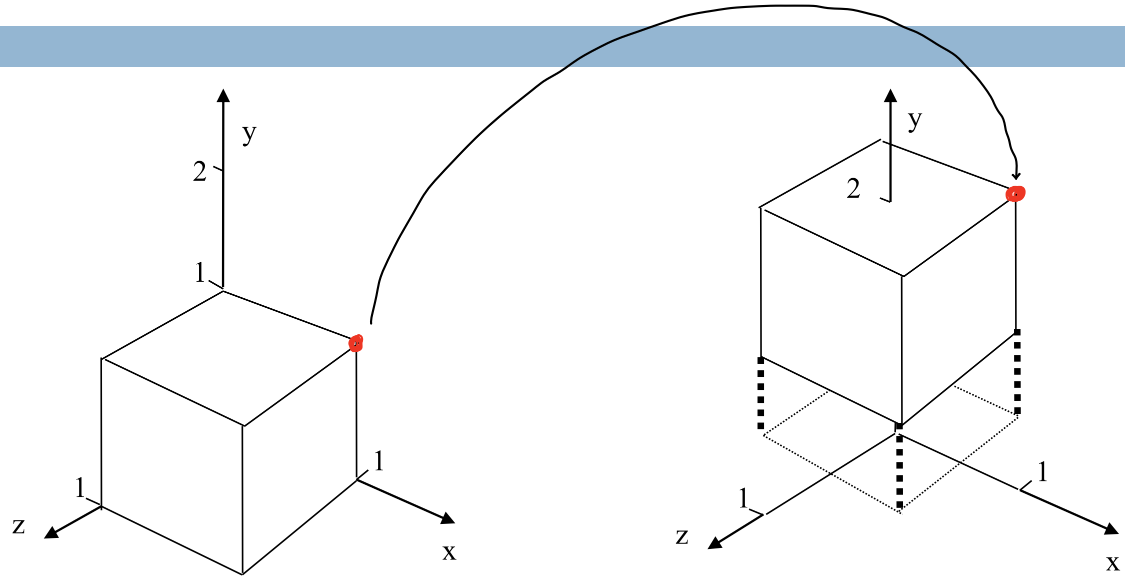
- To translate a point, just add its vector with the displacement vector (we will see later how to express it with a matrix-vector multiplication)

→ il problema è che noi vogliamo che tutte le trasformazioni siano MATRICE × VETTORE

- $$\begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} 1 \\ 3 \\ 0 \end{bmatrix} = \begin{bmatrix} x+1 \\ y+3 \\ z \end{bmatrix}$$

translate by 1 in the X direction and 3 in the Y direction

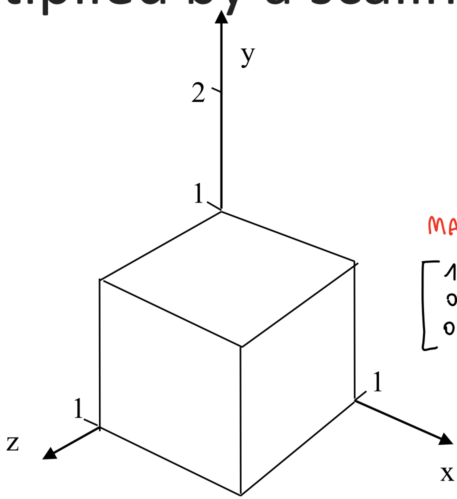
# Translate




- Add $\begin{bmatrix} -0.5 \\ 1 \\ -0.5 \end{bmatrix}$ to each vertex

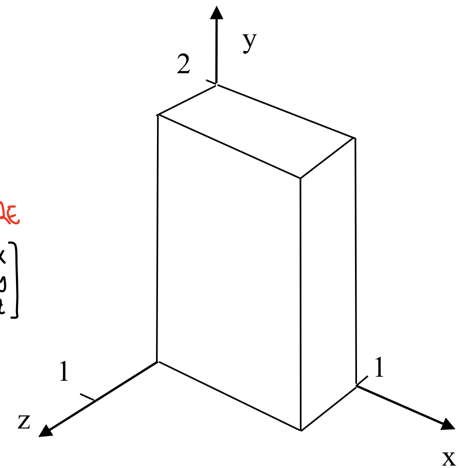
# Scale (at the origin)

- To scale a point, each element of its vector is multiplied by a scaling factor




MATRICE x VETTORE

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




- $\begin{bmatrix} x \\ y \\ z \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ 2\,y \\ 0.5\,z \end{bmatrix}$

=> NON PER FORZA devo avere lo stesso fattore di scala per tutte le 3 dimensioni

# Scale (at the origin)

- Scaling can be expressed as a matrix-vector multiplication

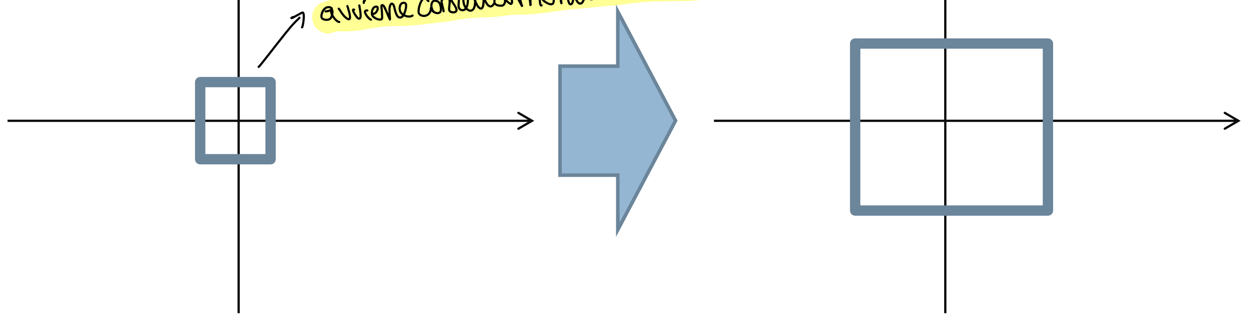
$$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} s_x x \\ s_y y \\ s_z z \end{bmatrix}$$

questi sono cambiamenti di scala ALL'ORIGINE (centro di scala 0,0,0)

↳ ovvero il punto che non cambia mai durante tutta la trasformazione è detto CENTRO DI SCALA, in questo caso è l'ORIGINE

# At the origin – what does it mean?

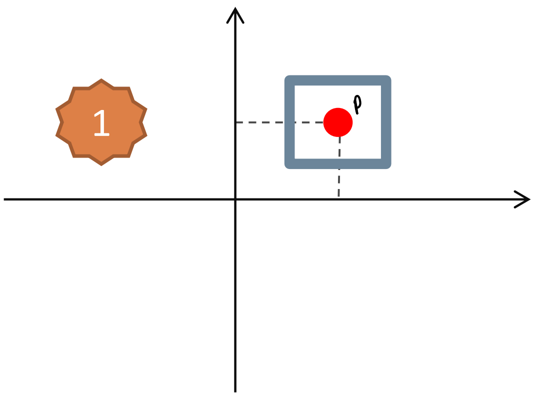
se l'ORIGINE è il CENTRO allora avviene correttamente il cambiamento di scala



se il CENTRO non è l'ORIGINE allora avviene, oltre che ad un cambiamento di scala, una traslazione

# Scaling around a generic point

considero P come il PUNTO del CENTRO di CAMBIAMENTO di SCALA

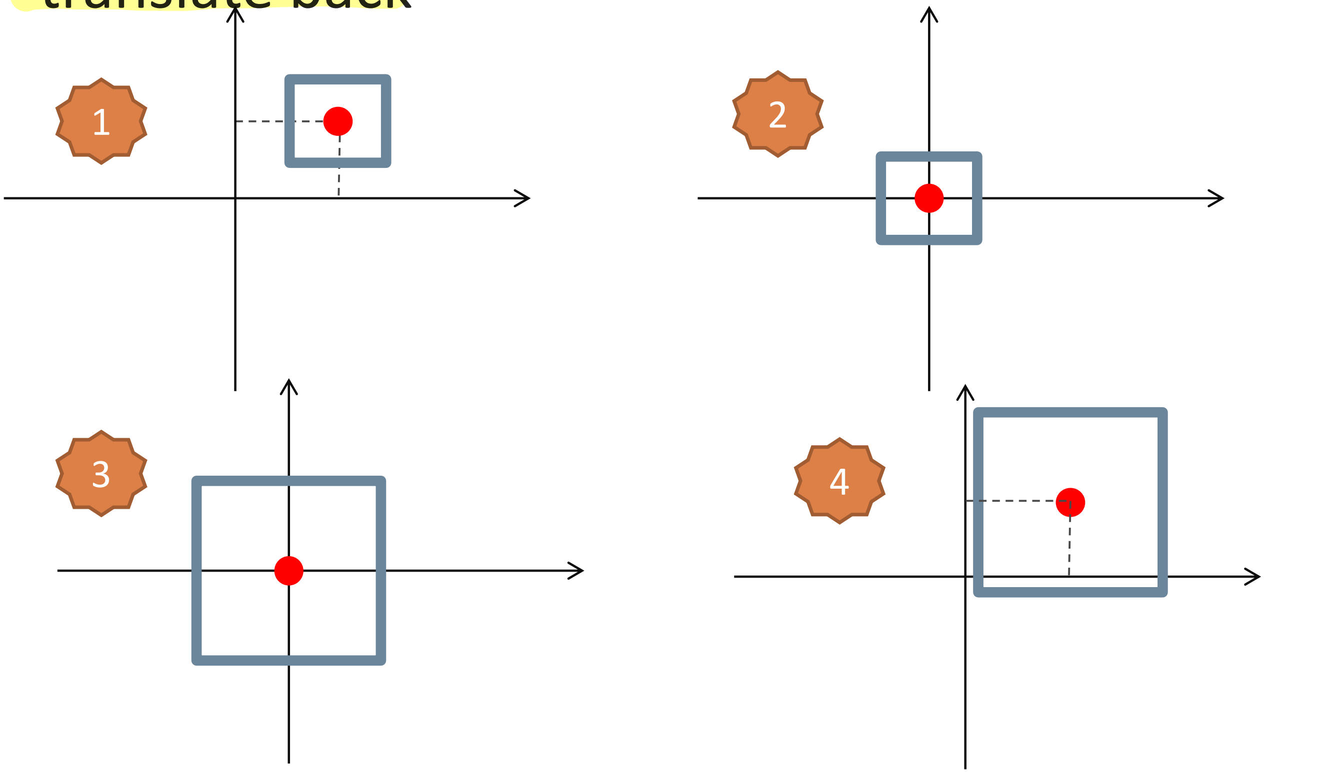



⇒

1. TRASLO tutto in modo che P coincida con l'ORIGINE
2. SCALO
3. RITRASLO nel punto iniziale (ovvero traslo tutto in modo che P torni nel punto di partenza)

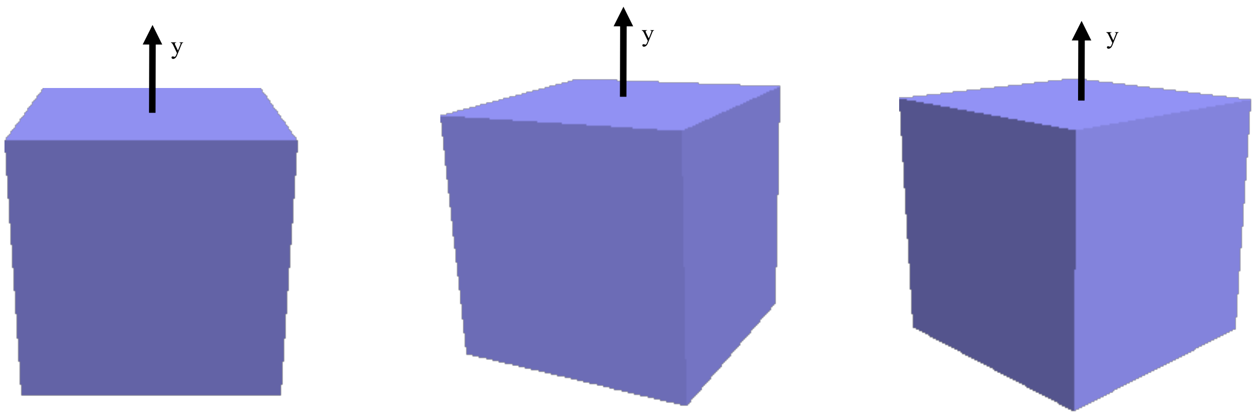
# Scaling around a generic point

- Translate the point at the origin, scale and translate back



# Rotations

- Three formulas, depending on the rotational axis




An example of rotation around the Y axis

ROTAZIONI : - intorno all'asse x
- intorno all'asse y
- intorno all'asse z

# Rotations

x non varia

- Around the x axis: $\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$

quando ruoto lungo l'asse x la x NON VARIA

$\Rightarrow [\dots] \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x \\ \dots \\ \dots \end{pmatrix}$

- Around the y axis: $\begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$

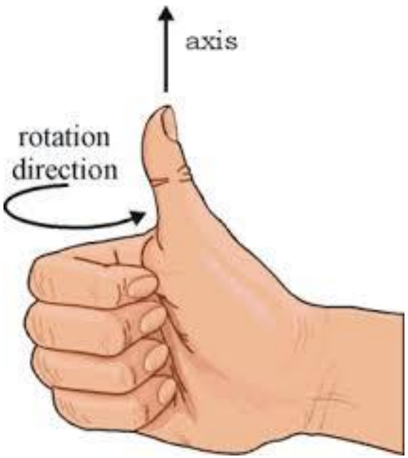
y non varia

- Around the z axis: $\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$

z non varia

# Direction of rotation

- Right hand rule. Put your thumb on the rotational axis. When you close the other fingers, they move in the direction of a positive rotation




(to rotate in the opposite direction, just use a negative angle…)

# Rotations

- Example. Rotate $\begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix}$ by 90° around the x axis

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 90° & -\sin 90° \\ 0 & \sin 90° & \cos 90° \end{bmatrix} \begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix} =$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 3 \\ -2 \\ 1 \end{bmatrix}$$

Note that x does not change

# Rotations

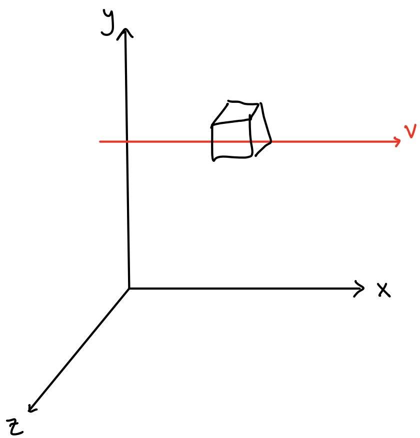
- Trigonometric table:

| | 0 | 30 | 45 | 60 | 90 | 180 | 270 | 360 |
|---|---|---|---|---|---|---|---|---|
| **cos** | 1 | $\sqrt{3}/2$ | $\sqrt{2}/2$ | 1/2 | 0 | -1 | 0 | 1 |
| **sin** | 0 | 1/2 | $\sqrt{2}/2$ | $\sqrt{3}/2$ | 1 | 0 | -1 | 0 |

# Rotations around generic axes

- Any rotation can be expressed as a combination of translations and "basic" rotations

Se voglio fare una rotazione su un asse qualsiasi (v)

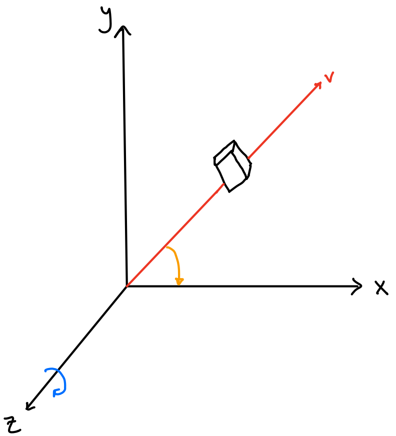



1. traslo v sull asse x in modo da farli coincidere
2. ruoto sull'asse x
3. annullo la traslaz. iniziale

# Rotations around generic axes

- Any rotation can be expressed as a combination of translations and “basic” rotations

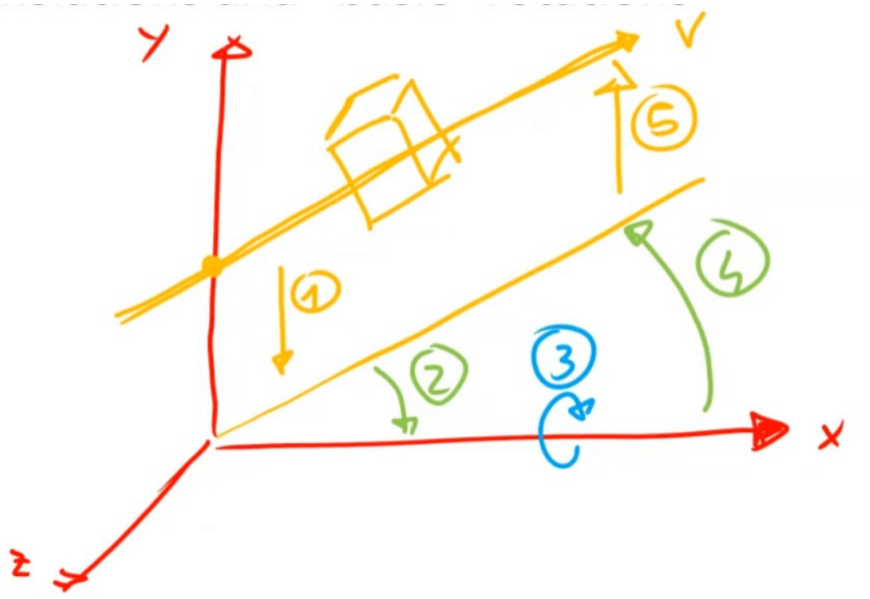
Se voglio fare una rotazione su un asse qualsiasi (v)




1. devo far ruotare v in modo da farlo coincidere sull'asse x • → ruoto z •
2. ruoto sull'asse x (che coincide con v)
3. annullo la rotaz. iniziale su z

# Rotations around generic axes

- Any rotation can be expressed as a combination of translations and "basic" rotations



# Combining transformations

non traslazioni

- Multiple consecutive rotations/scale changes can be easily expressed by a sequence of matrix multiplications:

$$\overset{\text{ROTAZ. z}}{\begin{bmatrix} \cos 45° & -\sin 45° & 0 \\ \sin 45° & \cos 45° & 0 \\ 0 & 0 & 1 \end{bmatrix}} \overset{\text{CAMBIO SCALA}}{\begin{bmatrix} 2 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{bmatrix}} \overset{\text{ROTAZ. x}}{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 90° & -\sin 90° \\ 0 & \sin 90° & \cos 90° \end{bmatrix}} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

in ordine da dx verso sx le scrivo, poi non conta l'ordine delle moltiplicazioni

A 90° rotation around the X axis, followed by a scale change, followed by a 45° Z axis rotation.

**Observation 1:** the "reading order" is from right to left

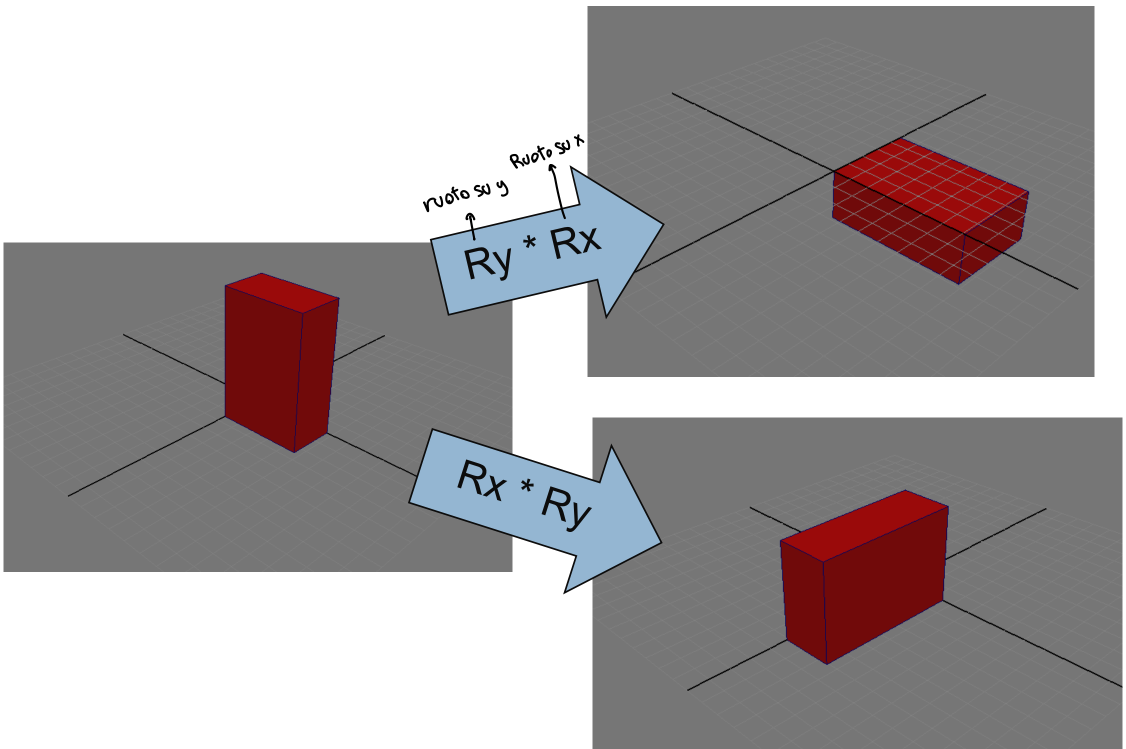
**Observation 2:** if you multiply the three matrices, you will obtain a **single** 3x3 matrix representing the three operations at once!

a JSRToolkit basta una sola matrice per localizzare il marker

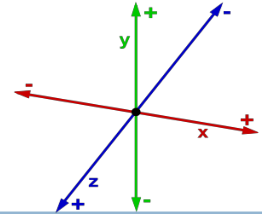
Thanks to the associative property of matrix multiplications

# Combining transformations

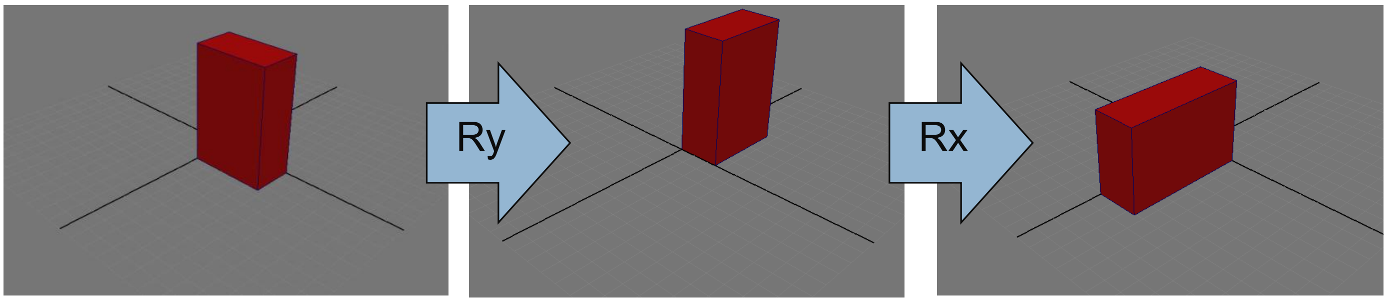
- We already know that matrix multiplication is **not** commutative. Consequence: the order of transformations **DOES** matter!



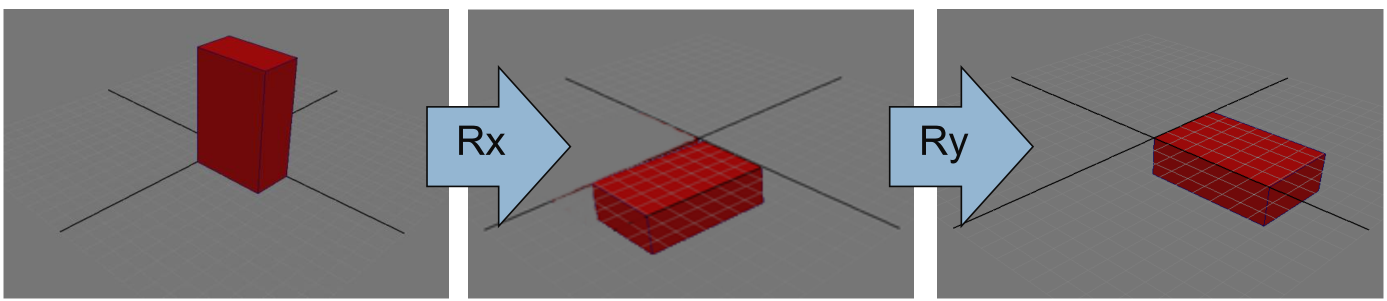
dx verso sx




Rx · Ry




Ry · Rx



# Combining transformations

- Rotations and scales can be easily combined in a single matrix using matrix multiplication
- It would be extremely useful if **ANY** transformation could be expressed as a single matrix

  → ovvero anche le TRASLAZIONI

- Unfortunately, translations (as seen up to now) cannot be expressed as a matrix

# Homogeneous coordinates

- With homogeneous coordinates, a point in the 3D space is represented by a **4D vector** according to the following rule:
- From non-homogeneous to homogeneous coordinates:

NON OM

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} hx \\ hy \\ hz \\ h \end{bmatrix} \qquad h \neq 0$$

OMOGENEO

- From homogeneous to non-homogenous coordinates:

$$\begin{bmatrix} x' \\ y' \\ z' \\ h' \end{bmatrix} = \begin{bmatrix} x'/h' \\ y'/h' \\ z'/h' \end{bmatrix}$$

(a full motivation of homogeneous coordinates is out of the scope of this course!)

# Examples

- Convert $\begin{bmatrix}1\\2\\3\end{bmatrix}$ from non-hom. to homogenous coordinates
- Many solutions exist:

$$\underset{\text{Non hom.}}{\begin{bmatrix}1\\2\\3\end{bmatrix}} = \underbrace{\begin{bmatrix}1\\2\\3\\1\end{bmatrix} = \begin{bmatrix}2\\4\\6\\2\end{bmatrix} = \begin{bmatrix}3\\6\\9\\3\end{bmatrix} = \begin{bmatrix}100\\200\\300\\100\end{bmatrix}}_{\text{homogenous}} = \dots$$

# Examples

- Convert $\begin{bmatrix} 9 \\ 3 \\ 12 \\ 3 \end{bmatrix}$ from hom. to non-homogeneous coords
- A single solution exists:

$$\begin{bmatrix} 9/3 \\ 3/3 \\ 12/3 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \\ 4 \end{bmatrix}$$

# Working with homogeneous coords

- Transformation matrices must be redefined
- Scale: CAMBIAMENTO DI SCALA con COORDINATE OMOGENEE

$$\begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ h \end{bmatrix} = \begin{bmatrix} s_x x \\ s_y y \\ s_z z \\ h \end{bmatrix}$$

# Rotations in hom. coordinates

- Around the x axis: $$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
- Around the y axis: $$\begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
- Around the z axis: $$\begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Translations in hom. coordinates

- Translation by $[t_x, t_y, t_z]$

elementi sulla diagonale sono 1

$$\begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} hx \\ hy \\ hz \\ h \end{bmatrix} = \begin{bmatrix} hx + ht_x \\ hy + ht_y \\ hz + ht_z \\ h \end{bmatrix}$$

→ e poi divido per h

⇒ ora so fare tutte le trasformazioni con una sola matrice

# Example

- Translate $\begin{bmatrix}1\\2\\3\end{bmatrix}$ by 3 in the X direction and 5 in the Y direction, using homogeneous coordinates

- First step: convert to homogeneous coordinates

$$\begin{bmatrix}1\\2\\3\end{bmatrix} \Rightarrow \begin{bmatrix}1\\2\\3\\1\end{bmatrix}$$

Infinite valid choices: use the simplest one!

# Example

- Second step: use the translation matrix

$$\begin{bmatrix} 1 & 0 & 0 & 3 \\ 0 & 1 & 0 & 5 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 1+3 \\ 2+5 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 7 \\ 3 \\ 1 \end{bmatrix}$$

- Third step: back to non-homogeneous coordinates:

$$\begin{bmatrix} 4 \\ 7 \\ 3 \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 4/1 \\ 7/1 \\ 3/1 \end{bmatrix} = \begin{bmatrix} 4 \\ 7 \\ 3 \end{bmatrix}$$

which is the expected result:

$$\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + \begin{bmatrix} 3 \\ 5 \\ 0 \end{bmatrix} = \begin{bmatrix} 4 \\ 7 \\ 3 \end{bmatrix}$$

# Final result

**With homogeneous coordinates, ANY possible transformation can be represented by a single $4 \times 4$ matrix**

# Example

- Write the matrix that translates an object by [1,2,0] and <u>then</u> rotates it around the X axis by 90°

(basically all the matrix exercises in the final examination will be like this)

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos 90 & -\text{sen}\, 90 & 0 \\ 0 & \text{sen}\, 90 & \cos 90 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

=> è IMPORTANTE l'ORDINE in cui scrivo le matrici ma l'ORDINE delle MOLTIPLICAZIONI NON CONTA!!

# Solution

- Translation >> homogeneous coords are required
- Translation matrix T: $\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
- Rotation matrix R: $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
- Final matrix: R·T = $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
- Be careful: it's R·T, ***not*** T·R