

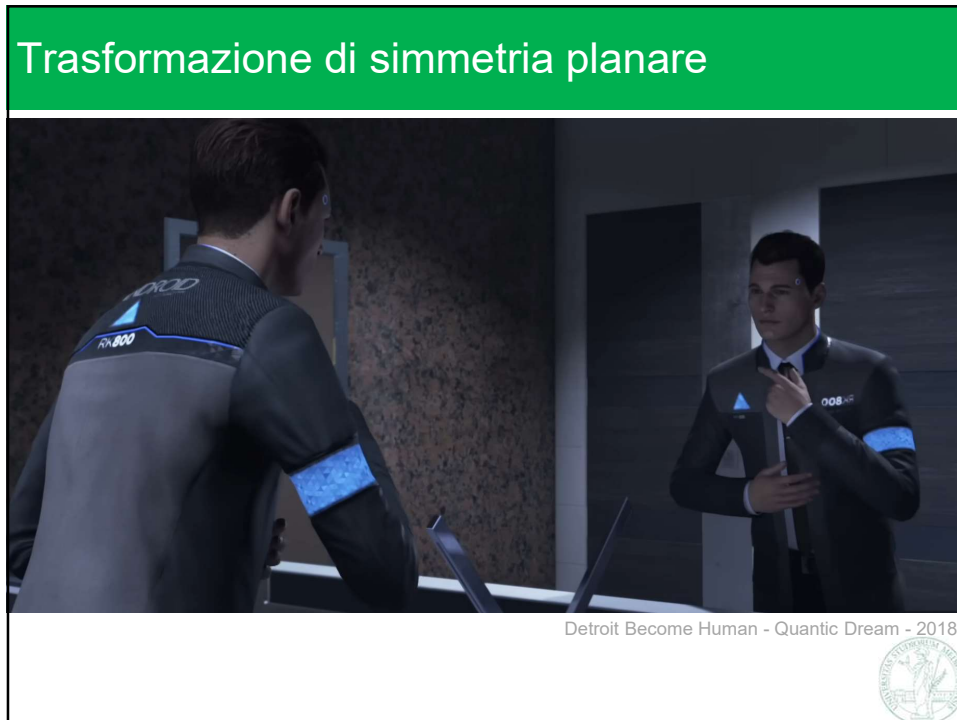
Marco Tarini - Computer Graphics 2025/2026
Università degli Studi di Milano

trasformazioni spaziali 2/2


The image is a slide from a presentation. At the top, a green header contains the text 'Marco Tarini - Computer Graphics 2025/2026' and 'Università degli Studi di Milano'. Below this, a white rounded rectangle with a green border contains the title 'trasformazioni spaziali 2/2'. The main content area shows a faint, large watermark of the University of Milan logo, which includes a woman holding a teapot. Overlaid on this watermark are two images: a wireframe teapot on the left and a photograph of an astronaut floating in space on the right.

37

Trasformazione di simmetria planare

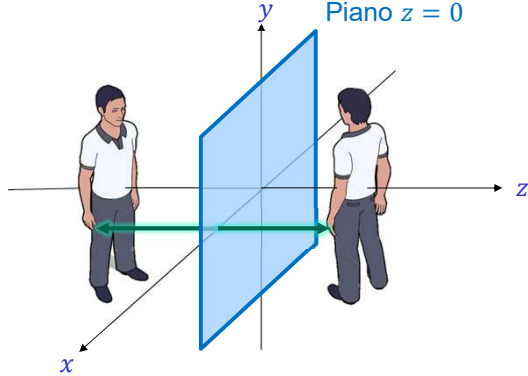



Detroit Become Human - Quantic Dream - 2018

The image is a slide from a presentation. At the top, a green header contains the title 'Trasformazione di simmetria planare'. Below this, a screenshot from the video game 'Detroit Become Human' shows two android characters in a futuristic setting. The character on the left is seen from the back, wearing a dark suit with blue glowing accents. The character on the right is facing him, also in a dark suit with blue glowing accents. The background is a dark, industrial-looking environment. Below the screenshot, the text 'Detroit Become Human - Quantic Dream - 2018' is visible. In the bottom right corner, there is a small circular logo of the University of Milan.

42

Matrice di simmetria planare (qui: per il piano $Z = 0$)




$$f \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x \\ y \\ -z \end{pmatrix} \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ -z \\ 1 \end{bmatrix}$$


43

Sommaro: esempi di trasformazioni affini esprese attraverso le loro matrici 4x4

$\begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$ <p>T_{t_x, t_y, t_z} matrice di Traslazione del vettore $\mathbf{t} = (t_x, t_y, t_z)$</p>	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ <p>M_{z} matrice di simmetria (Mirroring) del piano $z = 0$</p>
$\begin{bmatrix} \gamma_x & 0 & 0 & 0 \\ 0 & \gamma_y & 0 & 0 \\ 0 & 0 & \gamma_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ <p>S_{$\gamma_x, \gamma_y, \gamma_z$} matrice di Scaling anisotropico</p>	$\begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ <p>R_{$z, 90^\circ$} matrice di Rotazione di 90° attorno all'asse delle z</p>



44

Trasformazioni Affini

- ✓ Le caratteristiche spaziali di una trasformazione si riflettono nelle caratteristiche matematiche della matrice corrispondente.
- ✓ Ad esempio, il **determinante** della matrice ha una chiara interpretazione geometrica:
è il fattore di scala dei volumi della trasformazione
 - ⇒ Se prendo un modello 3D di una bottiglia da 2.5 litri e gli applico una trasformazione **M** (cioè trasformo tutti i suoi vertici), la bottiglia risultante avrà una capienza $2.5 \cdot \det(\mathbf{M})$ litri
 - ⇒ Se il determinante è negativo, l'oggetto si «ribalta» specularmente (le cose in *sensu orario* diventano antiorario, le scritte disegnate sopra si leggeranno *oinisrtuol s*, etc)
 - ⇒ Esercizio: verificare, trovando il determinante delle matrici viste fin'ora
- ✓ Il vantaggio principale di questa rappresentazione delle trasformazioni spaziali consiste nella loro composizione
 - ⇒ Vediamo in che modo



45

Eseguire sequenze di trasformazioni: note

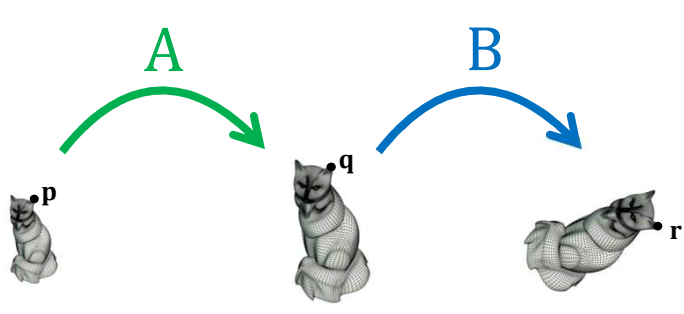
- ✓ Le trasformazioni affini sono chiuse per composizione.
...infatti...
- ✓ La moltiplicazione fra matrici / vettori è associativa
$$\mathbf{M}_B \cdot (\mathbf{M}_A \cdot \mathbf{p}) = (\mathbf{M}_B \cdot \mathbf{M}_A) \cdot \mathbf{p}$$
 - ⇒ La matrice \mathbf{M}_A «fa» una data trasformazione f_A
 - ⇒ La matrice \mathbf{M}_B «fa» una data trasformazione f_B
 - ⇒ La matrice $(\mathbf{M}_B \cdot \mathbf{M}_A)$ «fa» la trasformazione f_A , seguita da f_B

↑ ↑
DOPO PRIMA
- ✓ Cioè: due (oppure n) trasformazioni al prezzo di una!
 - ⇒ Moltiplicando per la matrice $\mathbf{M}_C = \mathbf{M}_B \cdot \mathbf{M}_A$, da sola, ottengo l'effetto che otterrei trasformando con \mathbf{M}_A e \mathbf{M}_B in cascata




46

Composizione di trasformazioni affini

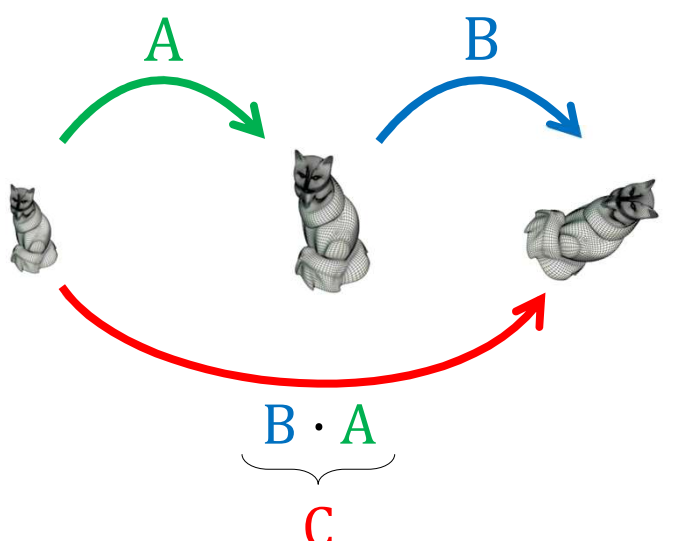


$\mathbf{q} = \mathbf{A} \cdot \mathbf{p}$
 $\mathbf{r} = \mathbf{B} \cdot \mathbf{q} = \mathbf{B} \cdot (\mathbf{A} \cdot \mathbf{p}) = \underbrace{(\mathbf{B} \cdot \mathbf{A})}_{\mathbf{C}} \mathbf{p}$




47

Composizione di trasformazioni affini



$\mathbf{B} \cdot \mathbf{A}$
 \mathbf{C}



48

Composizione di trasformazioni affini

The diagram illustrates the composition of three affine transformations, M_0 , M_1 , and M_2 , applied sequentially to a 3D model of a cat. M_0 is a translation, M_1 is a rotation, and M_2 is a scaling. The total transformation is represented by the matrix product $M_2 \cdot M_1 \cdot M_0$, labeled as M_{tot} .

49

Ripasso: moltiplicazione fra matrici (cioè composizione di trasformazioni)

- ✓ Date tre matrici quadrate A , B , C :
- ✓ Associativa: $A(BC) = (AB)C$
- ✓ Ma non commutativa! $AB \neq BA$

⇒ previsione:
determinare il corretto ordine delle trasformazioni non sarà intuitivo

The diagrams illustrate the non-commutativity of matrix multiplication. The left diagram shows a translation (RT) followed by a rotation. The right diagram shows a rotation (TR) followed by a translation.

50

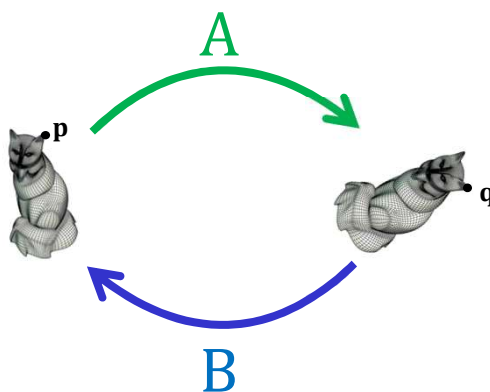
Esercizio

- ✓ Comporre (moltiplicare, nell'ordine giusto) una matrice di rotazione di 90 gradi attorno alle z con una matrice di traslazione del vettore (1,2,3) per ottenere la matrice che "ruota-poi-trasla"
⇒ Hai trovato così una matrice di "roto-traslazione"
- ✓ Trovare poi la matrice che ottegni se invece *trasla poi ruota*
⇒ È la stessa matrice? In cosa differisce dalla prima?



51

Inversa di una trasformazione



$$B = A^{-1}$$

$$q = A \cdot p$$

$$p = B \cdot q$$



52

Calcolo dell'inversa di una matrice di trasformazione affine

- ✓ Potremmo applicare un algoritmo generico di inversione di matrici 4x4
- ✓ molte delle matrici di trasformazioni utili sono però casi particolari che ci consentono di ottenere in modo semplificato (ed efficiente) l'inversa
- ✓ Vediamo per esempio l'inversa delle matrici di: traslazione, scalatura, rotazione, e di composizione di matrici



53

Inversa della matrice di scalatura

matrice di scalatura: $\mathbf{s}_{\gamma_x, \gamma_y, \gamma_z} = \begin{bmatrix} \gamma_x & 0 & 0 & 0 \\ 0 & \gamma_y & 0 & 0 \\ 0 & 0 & \gamma_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

l'inversa è, logicamente:

$$(\mathbf{s}_{\gamma_x, \gamma_y, \gamma_z})^{-1} = \mathbf{s}_{1/\gamma_x, 1/\gamma_y, 1/\gamma_z} = \begin{bmatrix} 1/\gamma_x & 0 & 0 & 0 \\ 0 & 1/\gamma_y & 0 & 0 \\ 0 & 0 & 1/\gamma_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



54

Inversa della matrice di traslazione

matrice di
 traslazione: $\mathbf{T}(\alpha_x, \alpha_y, \alpha_z) = \begin{bmatrix} 1 & 0 & 0 & \alpha_x \\ 0 & 1 & 0 & \alpha_y \\ 0 & 0 & 1 & \alpha_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$

l'inversa è, logicamente:

$$\mathbf{T}^{-1}(\alpha_x, \alpha_y, \alpha_z) = \mathbf{T}(-\alpha_x, -\alpha_y, -\alpha_z) = \begin{bmatrix} 1 & 0 & 0 & -\alpha_x \\ 0 & 1 & 0 & -\alpha_y \\ 0 & 0 & 1 & -\alpha_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



55

Inversa di matrice di simmetria speculare

matrice di
 rotaz di 90°
 in senso **antiorario**
 attorno
 all'asse delle z: $\begin{bmatrix} 0 & -1 & 0 & 0 \\ +1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

l'inversa è, logicamente...

matrice di
 rotaz di 90°
 in senso **orario**
 attorno
 all'asse delle z: $\begin{bmatrix} 0 & +1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Nota: la matrice inversa è anche la trasposta.
 Come vedremo, questo varrà per tutte le matrici di rotazione.



56

La matrice 4x4 corrispondente ad una trasformazione affine: sommario

Questa sottomatrice **M** 3x3 si applica alle coord cartesiane sia dei **punti** che dei **vettori**

Ultima riga: (0,0,0,1)
Così che i **punti** vengano mappati sempre in **punti**, e i **vettori** sempre in **vettori**

Questo vettore **t** si somma al risultato quando trasformo **punti** ma viene annullato (moltiplicato per zero) quando trasformo **vettori**.

M	t
0	1

Matrice 4x4

57

Alcune caratteristiche numeriche delle matrici 4x4 e loro interpretazione geometrica

- ✓ Matrice inversa = trasformazione inversa
 - ⇒ se esiste (se $\det \neq 0$)
- ✓ Determinante della matrice 4x4 = (che è anche il determinante della sottomatrice 3x3)
 - ⇒ Moltiplicatore del volume
 - ⇒ Per es, determinante = 0.5: questa matrice riduce i volumi degli oggetti trasformati del 50%
- ✓ Deficit di rango della matrice:
 - ⇒ cioè rango massimo (4) meno rango della matrice
 - ⇒ È la perdita di dimensionalità
 - ⇒ Per es: rango 3 (invece di 4): la trasf. riduce oggetti 3D a oggetti piatti.
 - ⇒ Per es: rango 2 (invece di 4): la trasf. riduce oggetti 3D ad una linea
 - ⇒ Per es: rango 1 (invece di 4): la trasf. riduce oggetti 3D ad un punto
 - ⇒ Se rango < 4: $\det = 0$ (il volume scompare) e la matrice non è invertibile (è «singolare»)


58

Digressione:
 in 2D: le trasformazioni affini sono matrici 3x3

✓ Punti 2D:

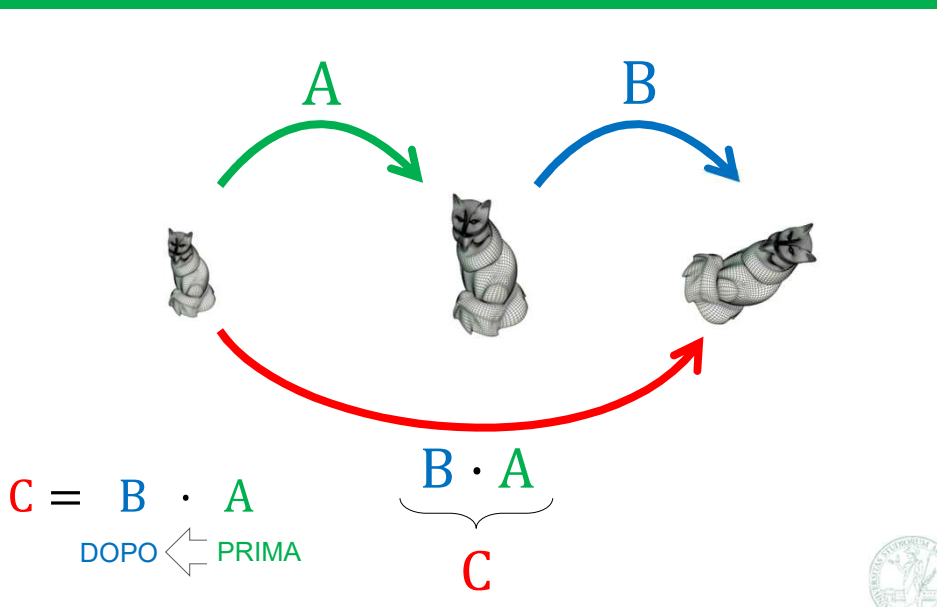
$$\begin{bmatrix} * & * & * \\ * & * & * \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix}$$

✓ Vettori 2D:

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


59


Composizione di trasformazioni affini (di nuovo)



$C = B \cdot A$

DOPO ← PRIMA

$B \cdot A$
 C



60

Inversa di una composizione di trasformazioni affini

✓ Attenzione all'inversione: $(B \cdot A)^{-1} = A^{-1} \cdot B^{-1}$

61

Trasf. di rotazione in 2D generica (attorno all'origine, di un angolo β generico, in senso antiorario)

preliminare: coordinate polari
 (un diverso modo per esprimere punti / vettori in 2D)

$\rho_0 = \sqrt{x_0^2 + y_0^2}$
 $\alpha_0 = \text{atan2}(y_0, x_0)$

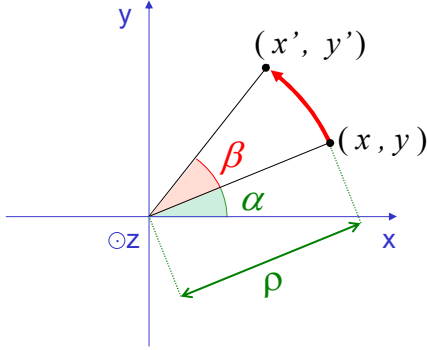
x_0, y_0
 coordinate cartesiane

$x_0 = \rho_0 \cdot \cos(\alpha_0)$
 $y_0 = \rho_0 \cdot \sin(\alpha_0)$

62


Trasf. di rotazione in 2D
 (attorno all'origine, di un angolo β **generico**, in senso antiorario)

In coordinate polari,
 la rotazione sarebbe banale:
 l'angolo α incrementa di β ,
 la distanza ρ rimane invariata



partenza:
 $x = \rho \cos \alpha$
 $y = \rho \sin \alpha$

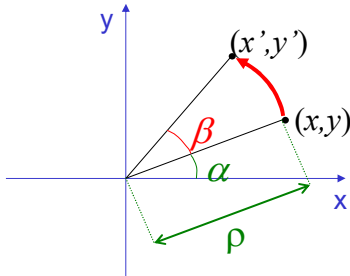
arrivo:
 $x' = \rho \cos(\alpha + \beta) = \rho \cos \alpha \cos \beta - \rho \sin \alpha \sin \beta = x \cos \beta - y \sin \beta$
 $y' = \rho \sin(\alpha + \beta) = \rho \cos \alpha \sin \beta + \rho \sin \alpha \cos \beta = x \sin \beta + y \cos \beta$



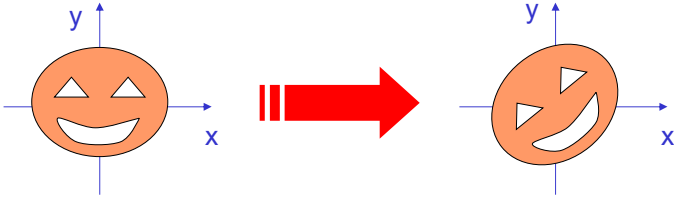

63

Trasf. di rotazione in 2D
 (attorno all'origine, di un angolo β **generico**, in senso antiorario)

Quindi:

$$f \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x \cos \beta - y \sin \beta \\ x \sin \beta + y \cos \beta \end{pmatrix}$$


alla fine, il passaggio alle coord polari
 serve solo per derivare la formula :-)
 Per applicarla, bastano il valore del seno
 e coseno dell'angolo di rotazione β

64

Trasf. di rotazione 3D attorno all'asse z (di un angolo β)

$$f \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x \cos \beta - y \sin \beta \\ x \sin \beta + y \cos \beta \\ z \end{pmatrix}$$

z rimane invariata
solo x e y ruotano

questo simbolo rappresenta un vettore / asse visto da davanti

67

Trasf. di rotazione 3D attorno all'asse z (di un angolo β)

$$f \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x \cos \beta - y \sin \beta \\ x \sin \beta + y \cos \beta \\ z \end{pmatrix}$$

Nota: questa trasformazione ruota attorno all'origine (non certo attorno al centro delle figure)

68

Rotazione 3D attorno all'asse z espresso come moltiplicazione con matrice

$$\begin{aligned} x' &= x \cos \beta - y \sin \beta \\ y' &= x \sin \beta + y \cos \beta \\ z' &= z \end{aligned}$$

matrice
di rotazione
di angolo β
attorno
all'asse Z

$$\begin{bmatrix} \cos \beta & -\sin \beta & 0 & 0 \\ \sin \beta & \cos \beta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \cos \beta - y \sin \beta \\ x \sin \beta + y \cos \beta \\ z \\ 1 \end{bmatrix}$$

70

Trasf. di rotazione attorno ad uno dei tre assi

Attorno ad
ASSE X

Attorno ad
ASSE Y

Attorno ad
ASSE Z

$$f \begin{pmatrix} x \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} x \\ y \cos \beta - z \sin \beta \\ y \sin \beta + z \cos \beta \end{pmatrix}$$

$$f \begin{pmatrix} x' \\ y \\ z' \end{pmatrix} = \begin{pmatrix} +z \cos \beta + x \sin \beta \\ y \\ -z \sin \beta + x \cos \beta \end{pmatrix}$$

$$f \begin{pmatrix} x' \\ y' \\ z \end{pmatrix} = \begin{pmatrix} x \cos \beta - y \sin \beta \\ x \sin \beta + y \cos \beta \\ z \end{pmatrix}$$

71

Matr di Rotazione attorno all'asse x , y , o z

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


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

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

e le inverse?

$$R_x(\theta)^{-1} = R_x(-\theta) = R_x(\theta)^T$$


verificare, ricordando che
 $\sin(-x) = -\sin(x)$ e $\cos(x) = \cos(-x)$



72

Generalizzare a qualsiasi matrice di rotazione

- ✓ Abbiamo visto solo rotazioni attorno ai tre assi X,Y,Z
- ✓ Possiamo però ottenere matrici di rotazioni attorno ad assi di rotazione *qualsiasi*:
 - ⇒ Le matrici di rotazione attorno ad un asse di rotazione qualsiasi (anche obliquo) purché *passante per l'origine* (cioè «pure») possono essere costruire cumulando in cascata (cioè, moltiplicando le matrici di) tre rotazioni sui tre assi X, Y, Z, con tre angoli opportunamente scelti (non vediamo come scegliere questi angoli)
 - ⇒ Le matrici di rotazione per un asse *non* passante per l'origine possono essere ottenute combinando (moltiplicando) rotazioni e traslazioni.
Vediamo come fra poco!



73

Alcune considerazioni sulle rotazioni pure in 3D (caso generale)

- ✓ In 3D, posso specificare una rotazione pura specificando attorno a quale **asse** di rotazione voglio ruotare
 - ⇒ E, di quale angolo (espresso in gradi o radianti)
 - ⇒ Convenzione sul segno: positivo se senso antiorario, negativo se in senso orario (guardando l'asse dalla punta verso la coda)
- ✓ La trasformaz si applica nello stesso modo ai punti e ai vettori
 - ⇒ la stessa funzione viene applicata alle loro coordinate
- ✓ Pura = che lascia l'origine invariata
- ✓ I punti sull'asse vengono mappati su se stessi
 - ⇒ e così, i vettori paralleli all'asse
- ✓ **Attenzione: cumulare rotazioni in 3D non commuta!**
 - ⇒ Es: ruotare attorno a X di 90° e poi a Y di 90°
non è la stessa cosa di
ruotare attorno a Y di 90° e poi a X di 90°



74

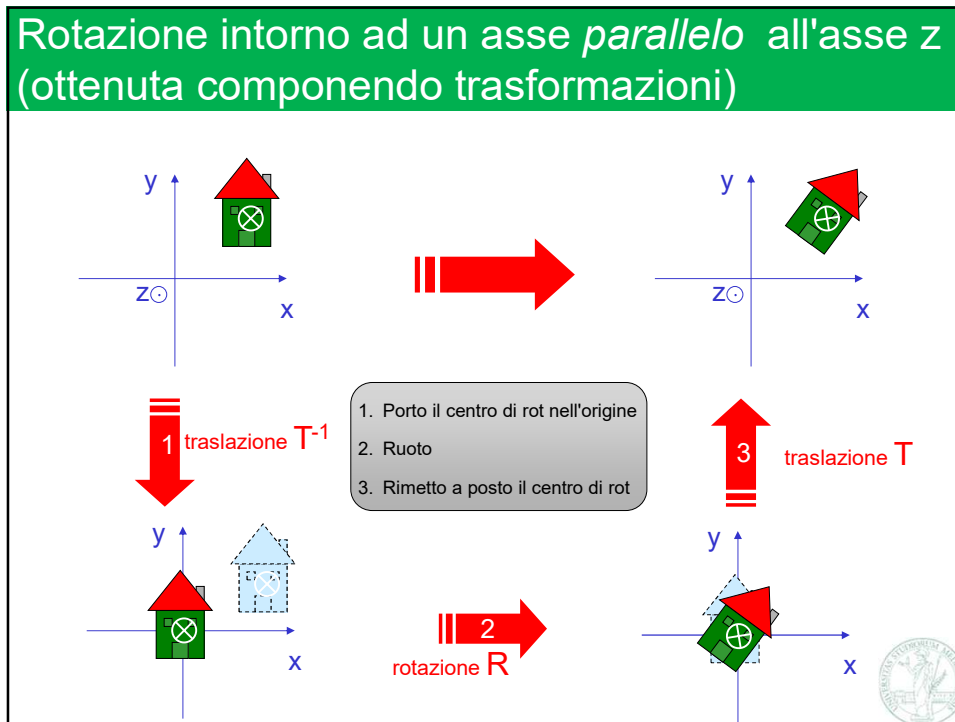
Alcune considerazioni sulle rotazioni in 3D (caso generale)

- ✓ Loro inversa: cambiare segno all'angolo $\beta = -\beta$
 - ⇒ cambiare il segno al $\sin \beta$ ($\cos \beta$ rimane invariato)
 - ⇒ la matrice si **traspone**
- ✓ L'**inversa** di una matrice di rotazione pura è la sua **trasposta**
- ✓ Sono dunque matrici «ortonormali» $M \cdot M^T = I$

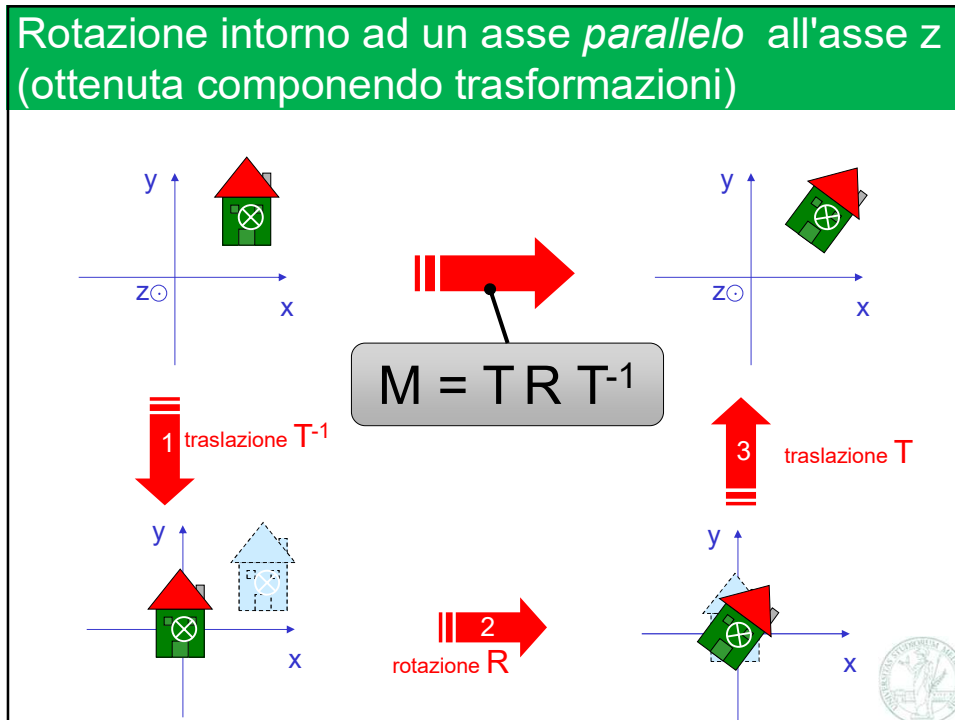
- ✓ E metrici di rotazione lungo assi non passante per l'origine?
 - ⇒ Posso ottnerere combinando rotazioni pure e traslazioni
 - ⇒ Vediamo con un esempio...



75



76



77

Rotazione intorno ad un asse *parallelo* all'asse z (ottenuta componendo trasformazioni)

✓ Moltiplicazione matrici (vettori) ha la proprietà associativa

$$f(p) = T (R (T^{-1} p))$$

$$= (TR T^{-1}) p$$

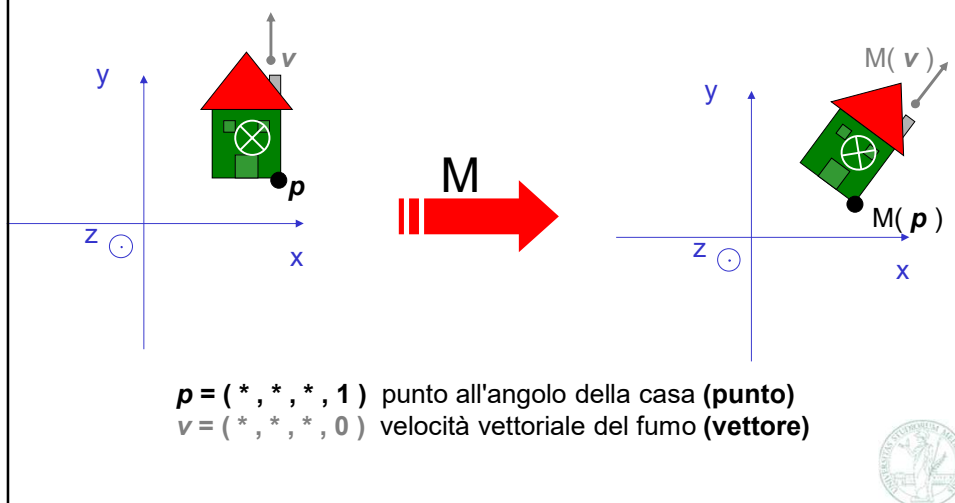
una matrice M 4x4
che fa tutto.

- considerazioni sull'efficienza
- cosa possiamo dire sulla forma di M ?
- cosa succede se moltiplichiamo un *vettore* per M ?



78

Punti VS vettori



79

Trasformazione di shearing

✓ Es: spostamento in x proporzionale alla pos y

$$x' = x + y \cot \theta$$

$$y' = y$$

$$z' = z$$

$$H_{xy}(\theta) = \begin{bmatrix} 1 & \cot \theta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

matrice di shearing
(della x rispetto alla y ,
di angolo θ)

82

Trasformazione di shearing

✓ Es: spostamento in x proporzionale alla pos y

$$x' = x + y \cot \theta$$

$$y' = y$$

$$z' = z$$

$$H_{xy}(\theta) = \begin{bmatrix} 1 & \cot \theta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

matrice di shearing
(della x rispetto alla y ,
di angolo θ)

83

Lista completa degli effetti geometrici ottenibili con trasformazioni affini

- ✓ Rotazioni
⇒ di asse e angolo qualsiasi
- ✓ Traslazioni
- ✓ Scalature
⇒ uniformi o no
⇒ compreso ribaltamenti
- ✓ Shearing
- ✓ ... e tutte le composizioni di queste operazioni

84

GUI comuni per specificare trasformazioni affini in 2D (questo esempio: da inkscape)

[DEMO]

(più la traslazione con drag-and-drop)
comulando trasformazioni, si possono ottenere tutte le possibili trasf affini in 2D

85



86

Trasformazioni spaziali affini: una definizione equivalente, e una proprietà

✓ Sono tutte e sole le funzioni **lineari** ,
 cioè tali che,

dati un punto \mathbf{p} $f(\mathbf{p} + k\vec{v}) = f(\mathbf{p}) + kf(\vec{v})$
 vettore \vec{v} , \vec{w}

e uno scalare k : $f(h\vec{v} + k\vec{w}) = h f(\vec{v}) + k f(\vec{w})$


- ⇒ Cioè: trasformare una combinazione lineare di punti (o vettori) (interpolazione compresa) è la stessa cosa di combinare linearmente i punti (o i vettori) trasformati
- ⇒ Questo è cruciale per la nostra applicazione (rendering basato su rasterizzazione), perché significa che...
 ... per applicare una trasformazione affine ad un triangolo basta applicarla ai suoi vertici, e poi congiungere i vertici trasformati
 ... per applicare una trasformazione ad una spline basta calcolare le trasformazioni dei punti di controllo, e poi usarli per la spline trasformata



88


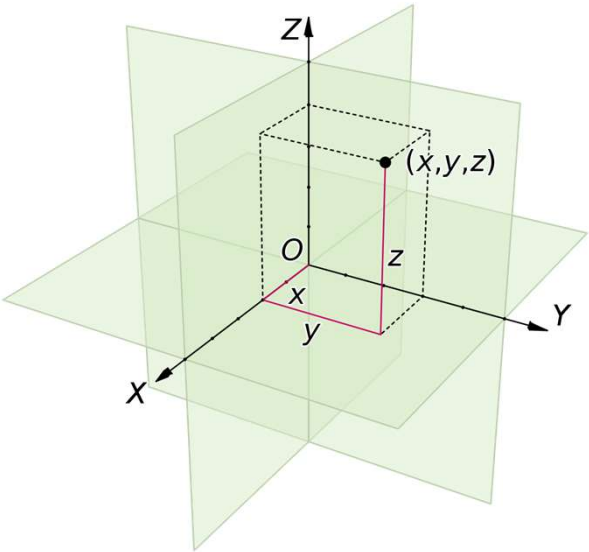
Trasformazioni affini come cambi di Sistema di riferimento

- ✓ Fin'ora, abbiamo interpretato le **trasformazioni affini** come azioni che vengono **applicate fisicamente ad oggetti**
 - ⇒ Ad esempio, li spostano, li riorientano nello spazio, li ingrandiscono o riducono, li deformano, etc.
- ✓ Vediamo ora un'utilissima interpretazione alternativa ma matematicamente equivalente, di cosa sia una trasformazione affine (e cosa significhi applicarla):
- ✓ Applicare una trasformazione affine ad un punto / un vettore può essere vista come la *conversione* delle coordinate del punto / vettore da un Sistema di Riferimento ad un altro



91

Ripasso: sistema di riferimento (o anche "spazio") in cui esprimere punti e vettori



92

Sistema di riferimento: note

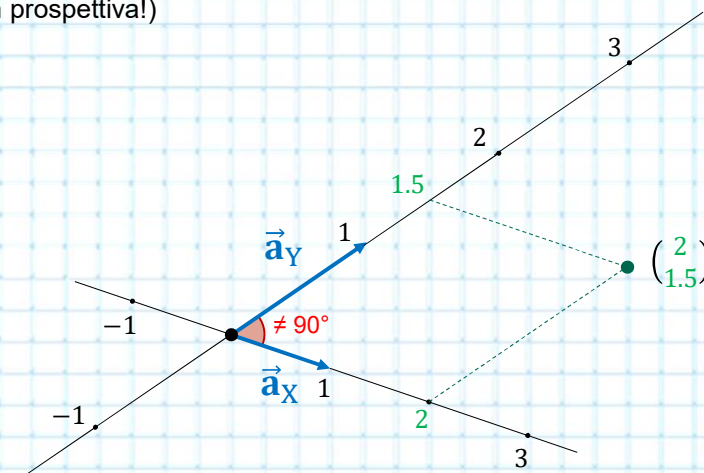
- ✓ Fin'ora, abbiamo tacitamente assunto che il Sistema di riferimento (che usiamo per esprimere sia punti che vettori) fosse quello **canonico**, cioè uno in cui:
 - ⇒ Gli assi sono ortogonali fra loro
 - ⇒ Gli assi hanno la stessa lunghezza, che è unitaria
 - ⇒ L'origine è in $(0,0,0)$
- ✓ Un **Sistema di Riferimento** in generale (un modo che ci consente di esprimere punti e vettori 3D come triplette di coordinate) può non avere nessuna di queste caratteristiche
 - ⇒ E' sufficiente che gli assi siano linearmente indipendenti, per garantire che ci sia un modo di esprimere qualsiasi punto o vettore



93

Un sistema di riferimento non ortogonale (in 2D)

(Questo disegno è in 2D, e non è in prospettiva!)



94

Sistema di riferimento o reference frame (oppure anche: uno spazio)

- ✓ Un sistema di riferimento R è definito da
 - ⇒ una base vettoriale $\{\vec{a}_x, \vec{a}_y, \vec{a}_z\}$ (gli assi dello spazio)
 - ⇒ un punto di *origine* \mathbf{p}_0
- ✓ Posso esprimere (univocamente!) ogni punto \mathbf{p} come:

$$\mathbf{p} = \mathbf{p}_0 + x\vec{a}_x + y\vec{a}_y + z\vec{a}_z$$
- ✓ Cioè, in forma matriciale:

coordinate
omogenee
di \mathbf{p}
nel sistema
di riferimento
canonico

$\left\{ \begin{array}{c} x' \\ y' \\ z' \\ 1 \end{array} \right\} =$

sistema di riferimento R

\vec{a}_x	\vec{a}_y	\vec{a}_z	\mathbf{p}_0
0	0	0	1

$=$

x
y
z
1

coordinate
omogenee
di \mathbf{p}
nel
sistema R

95

Base vettoriale (o spazio vettoriale)

- ✓ Definita da tre vettori asse (linearmente indipendenti)
 - ⇒ $\{\vec{a}_x, \vec{a}_y, \vec{a}_z\}$
- ✓ Posso esprimere (univocamente!) ogni vettore \vec{v} come:

$$\vec{v} = x\vec{a}_x + y\vec{a}_y + z\vec{a}_z$$
- ✓ Cioè:

coordinate
omogenee
di \vec{v}
nella base
canonica

$\left\{ \begin{array}{c} x' \\ y' \\ z' \\ 0 \end{array} \right\} =$

Base vettoriale B

\vec{a}_x	\vec{a}_y	\vec{a}_z	*
0	0	0	1

Non
conta

$=$

x
y
z
0

coordinate
omogenee
di \vec{v}
nella base del
sistema B

96

Ripasso: prodotto Matrice × Vettore
 («riga per colonna»)

✓ Posso scriverlo come 4 prodotti dot con i vettori riga

99

Ripasso: prodotto Matrice × Vettore
 («riga per colonna»)

✓ Ma posso anche scriverlo come:
 una **combinazione lineare** dei **vettori colonna**

100

Cambio di Sistema di riferimento (o spazio)

- ✓ Ogni trasformazione affine può essere interpretata come un *cambio di sistema di riferimento (o spazio)*
- ✓ e le 4 **colonne** della matrice riportano
 - ⇒ 3 assi
 - ⇒ originedello **spazio di partenza** descritte nelle coordinate dello **spazio di arrivo**



101

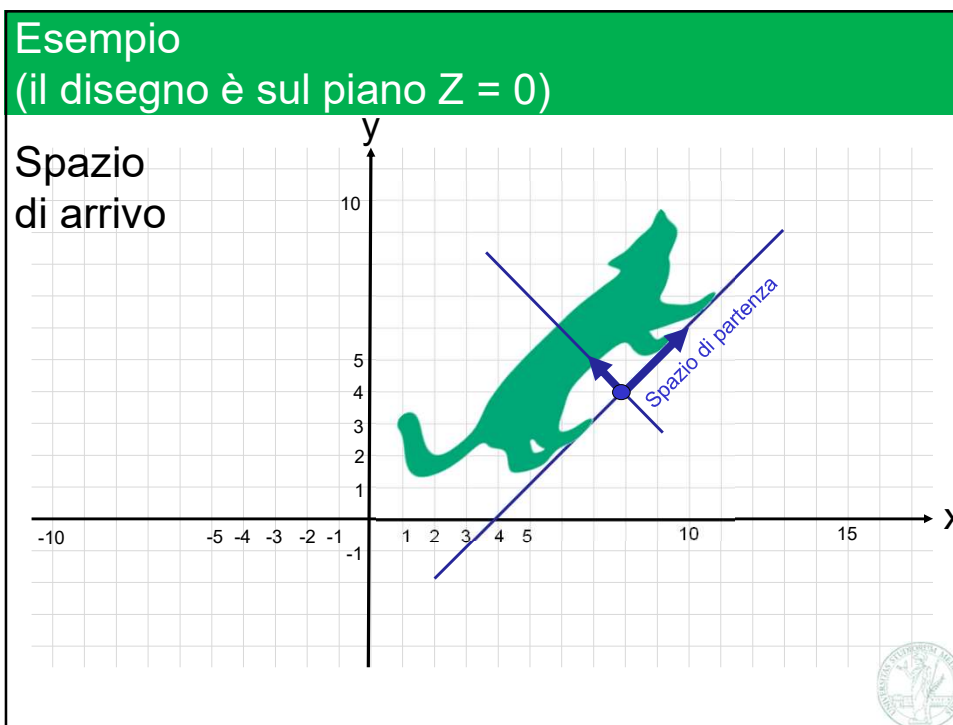
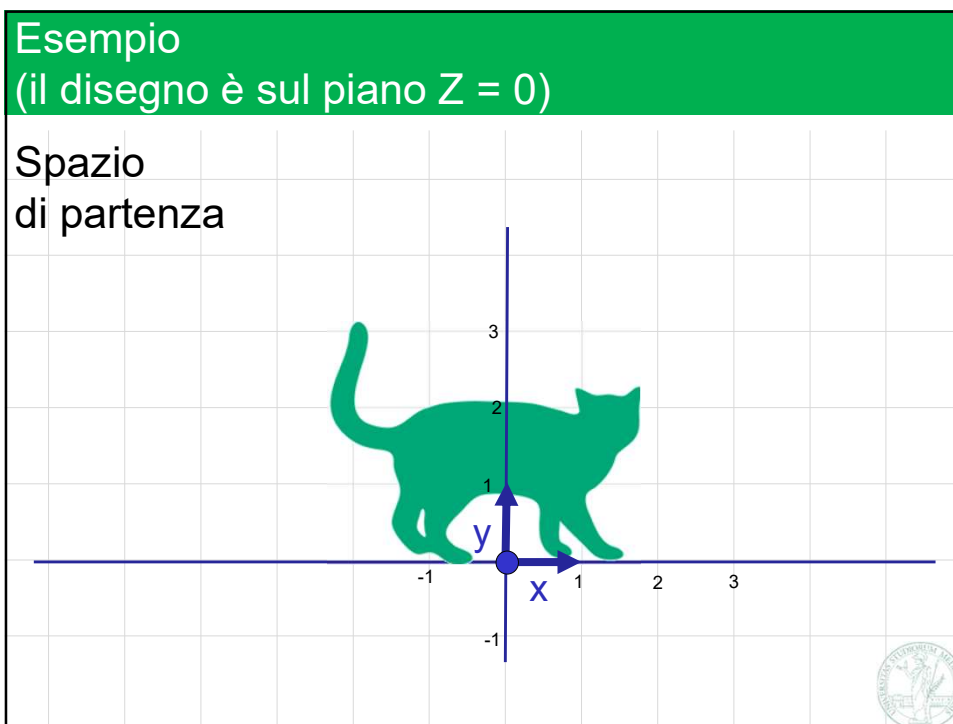
Sistema di riferimento canonico (quello che abbiamo tacitamente assunto finora)

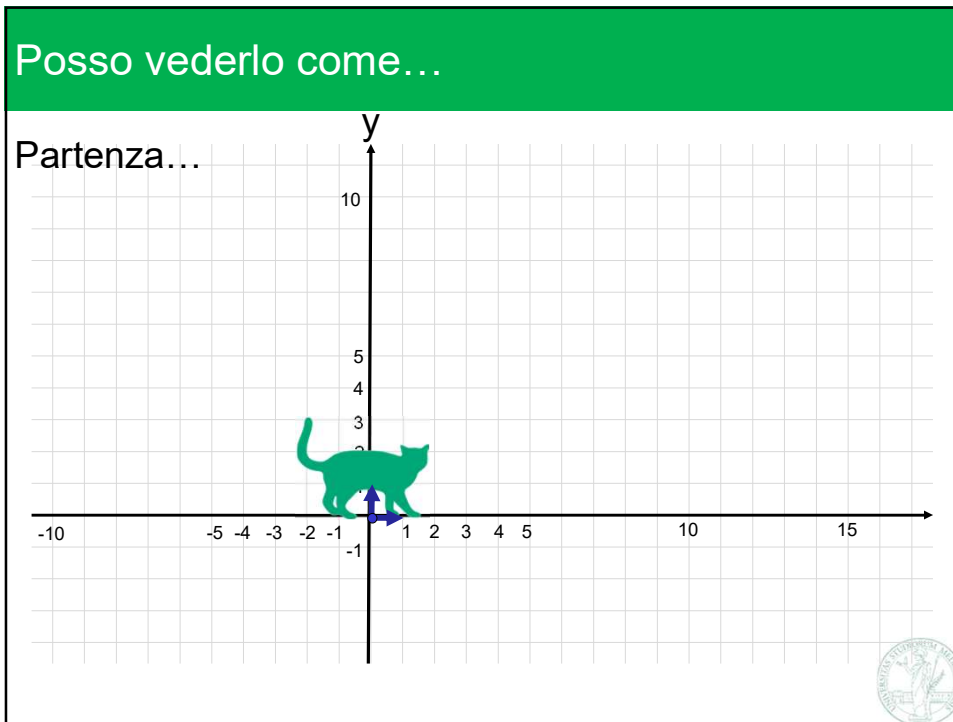
- ✓ Assi:
 $\vec{a}_x = (1,0,0)$
 $\vec{a}_y = (0,1,0)$
 $\vec{a}_z = (0,0,1)$
- ✓ Origine: $\mathbf{p}_0 = (0,0,0)$

- ✓ Quindi, matrice associata: $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = I$

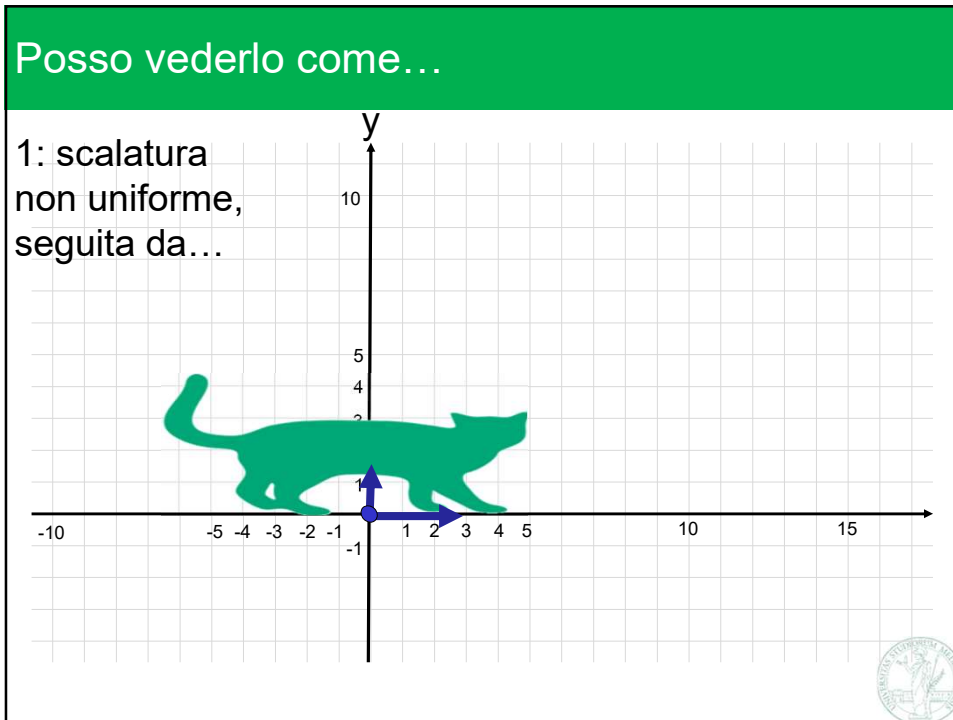


102

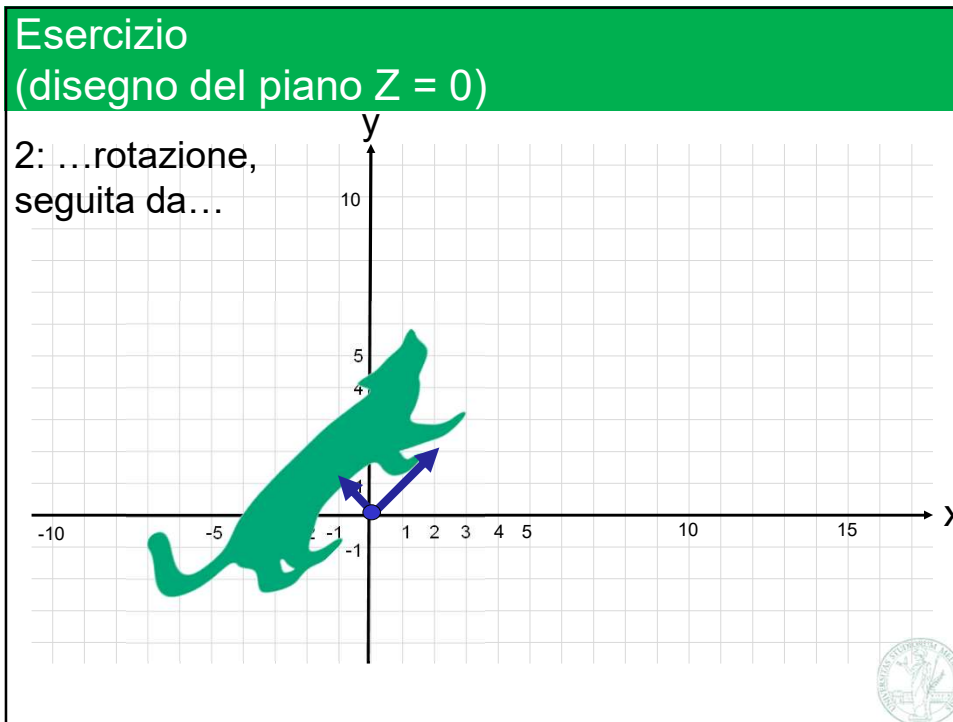




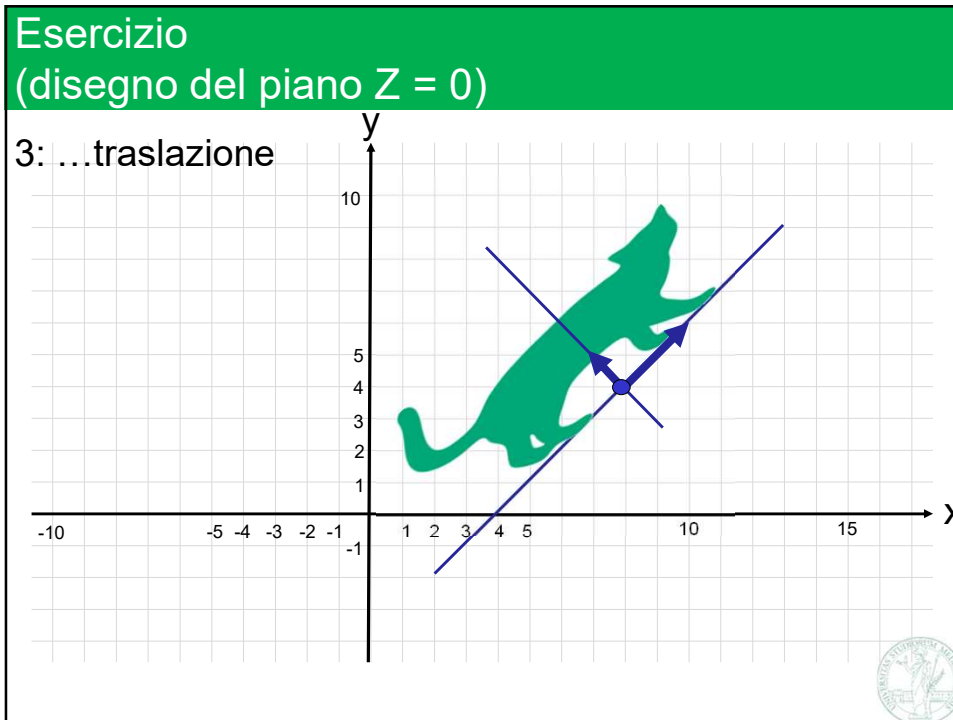
117



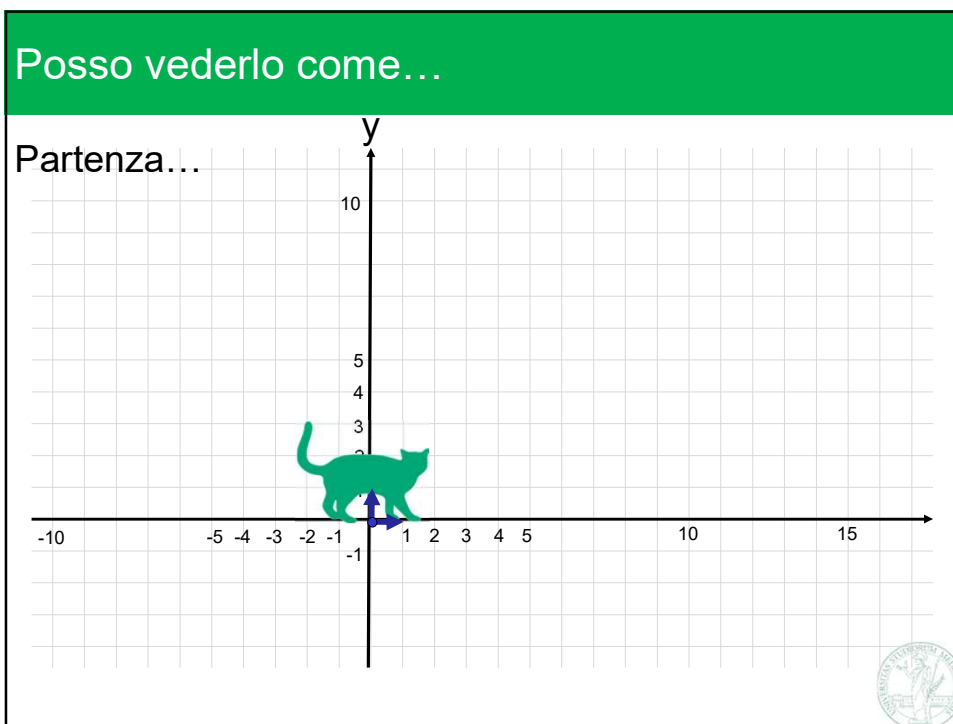
118



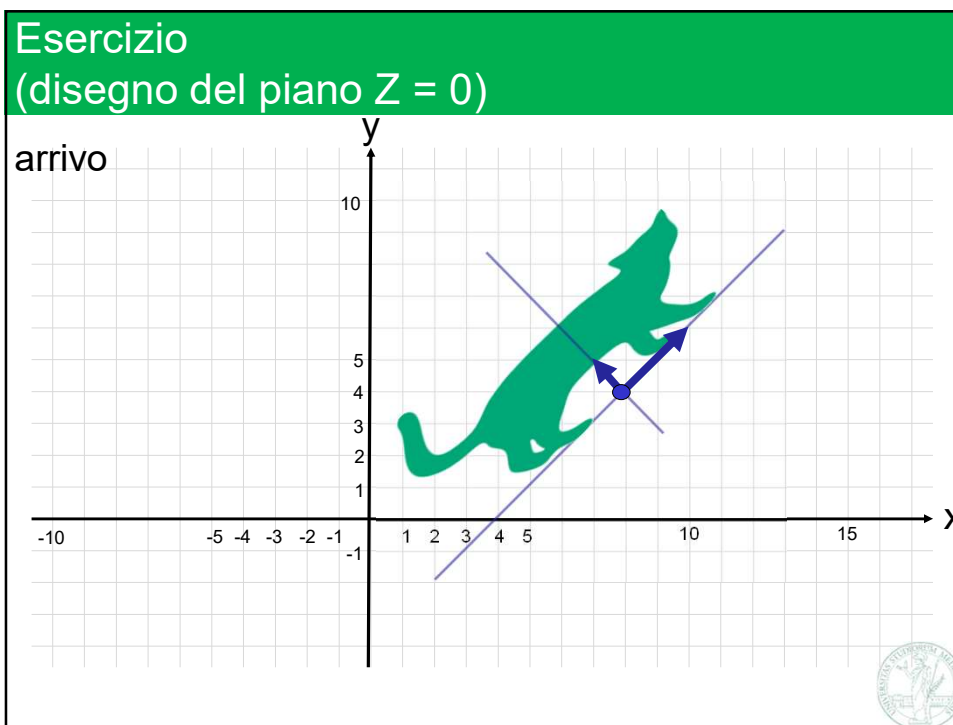
119



120



121



122

Note sull'esempio

- ✓ Per passare dallo spazio di partenza a quello di arrivo, abbiamo sottoposto il gatto ad una sequenza di trasformazioni spaziali
 - ⇒ Giudicando ad occhio, il gatto è stato... (in sequenza)
 1. scalato anisotropicamente (leggermente alzato verticalmente, e molto allungato orizzontalmente),
 2. poi, ruotato di 45° in senso ccw attorno asse z
 3. poi, traslato
- ✓ Sarebbe possibile ottenere la matrice di trasformazione risultante moltiplicando le 3 matrici associate a ciascuna delle trasformazioni T1, T2, T3 (da destra a sinistra), ottenendo come risultato una matrice complessiva M
- ✓ La matrice M applica in un colpo solo tutte le trasformazioni
- ✓ (Potremmo anche immaginarci diverse sequenze di trasformazioni che ottengono l'effetto mostrato: il loro prodotto matriciale otterrebbe sempre la stessa matrice M)



125

Note sull'esempio

- ✓ Un metodo alternativo (equivalente, ma a volte più semplice, come in questo caso) di capire il valori della matrice M , è guardare direttamente il suo effetto complessivo, trascrivendo, nelle sue 4 colonne, i tre assi x,y,z (vettori) e l'origine (punto), del sistema di riferimento originale (in blu), espressi in coordinate affini (cioè cartesiane, e 0 o 1 come coordinata w) nel sistema di riferimento di arrivo (in nero).



126

Esercizio 1

1. Scrivi la matrice che porta dallo spazio di partenza mostrato allo spazio di arrivo mostrato
 - ⇒ Come: riporta nelle colonne della matrice le coordinate (omogenee) dei vettori asse, e del punto origine, del Sistema di riferimento di partenza, come li vedi nel Sistema di riferimento di arrivo
2. Applica la matrice trovata al punto \mathbf{p} , che nello spazio di partenza ha coordinate cartesiane $(-2,3,0)$
 - ⇒ come si vede dal primo disegno: è l'estremità della coda
3. *Hai ottenuto:* le coordinate \mathbf{p}' dello stesso punto nello spazio di arrivo
 - ⇒ Verifica sul secondo disegno



127

Esercizio 2

“Una mosca si poggia a coordinate (nel Sistema di riferimento di arrivo) $(9,8,0)$. In quale punto del Sistema di riferimento di partenza si trova la mosca?”

1. Inverti la matrice del punto precedente
 - ⇒ facendo alcuni conti, oppure avvalendoti di un software di appoggio, per es <https://matrix.reshish.com/inverse.php>)
2. Applica la matrice inversa al punto di coordinate che, nel Sistema di Riferimento di arrivo $(9,8,0)$
 - ⇒ *come si vede dal secondo disegno: corrisponde ad punto che è, circa, collocato sul naso del gatto*
3. Hai ottenuto le coordinate dello stesso punto nel Sistema di riferimento di partenza
 - ⇒ verifica sul primo disegno



128

Un modo di interpretare qualsiasi trasformazione affine: sommario

- ✓ In pratica, una trasformazione affine si limita a ridefinire un nuovo sistema di riferimento arbitrario
⇒ 3 assi (vettori) + origine (punto)
- ✓ L'oggetto 3D segue il nuovo sistema di riferimento

Oggetto 3D *qualsiasi* (mesh, campo d'altezza, sup parametrica...) espresso da punti/vettori nel **proprio** sistema di riferimento

Un nuovo sistema di riferimento qualsiasi (nuovi assi e nuova origine)

129

Un modo di interpretare qualsiasi trasformazione affine: sommario

- ✓ In pratica, una trasformazione affine si limita a ridefinire un nuovo sistema di riferimento arbitrario
⇒ 3 assi + origine
- ✓ L'oggetto 3D segue il nuovo sistema di riferimento

Lo stesso oggetto 3D (qui: 2D), ma ridefinito sul nuovo sist. di rif., cioè trasformato

Matrice che codifica il sist. di rif.

trasformazione affine

130