ТЕМА №19

Матрици

Съдържание

Тема 19: Матрици

- Матрици и геометрии
- Транслация
- Мащабиране
- Ротация

Матрици и геометрии

Употреба

Употреба на матрици

- Моделиране на трансформации (транслация, ротация, мащабиране)
- Анимация чрез матрици
- Проекция и перспектива
- Контрол на гледната точка

Матриците

Направо за 3D пространство

- 2D трансформациите са частни случаи
- Хомогенни координати (т.е. 4x4)

$$M = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix}$$

- Да припомним: точка $P(p_x, p_y, p_z)$ в хомогенни координати е $P(p_x, p_y, p_z, 1)$

Видове матрици

Базисни матрици

- Базисните трансформации и анимации се моделират с базисни матрици
- Сложните трансформации и анимации се моделират със съставни матрици

Получаване на съставни матрици

- Чрез умножение на базисни матрици
  $M = M_1 M_2 M_3 \dots M_n$

Единична матрица

– Запазва непроменен обекта

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

– Всички базисни матрици са леко изменени единични матрици

Характеристики

Преимущества

- Почти всичко се прави с матрици
- Лесно и еднотипно изчисляване
- Вместо няколко базисни матрици се ползва направо съставната им
- Налични са в много графични системи

Недостатъци

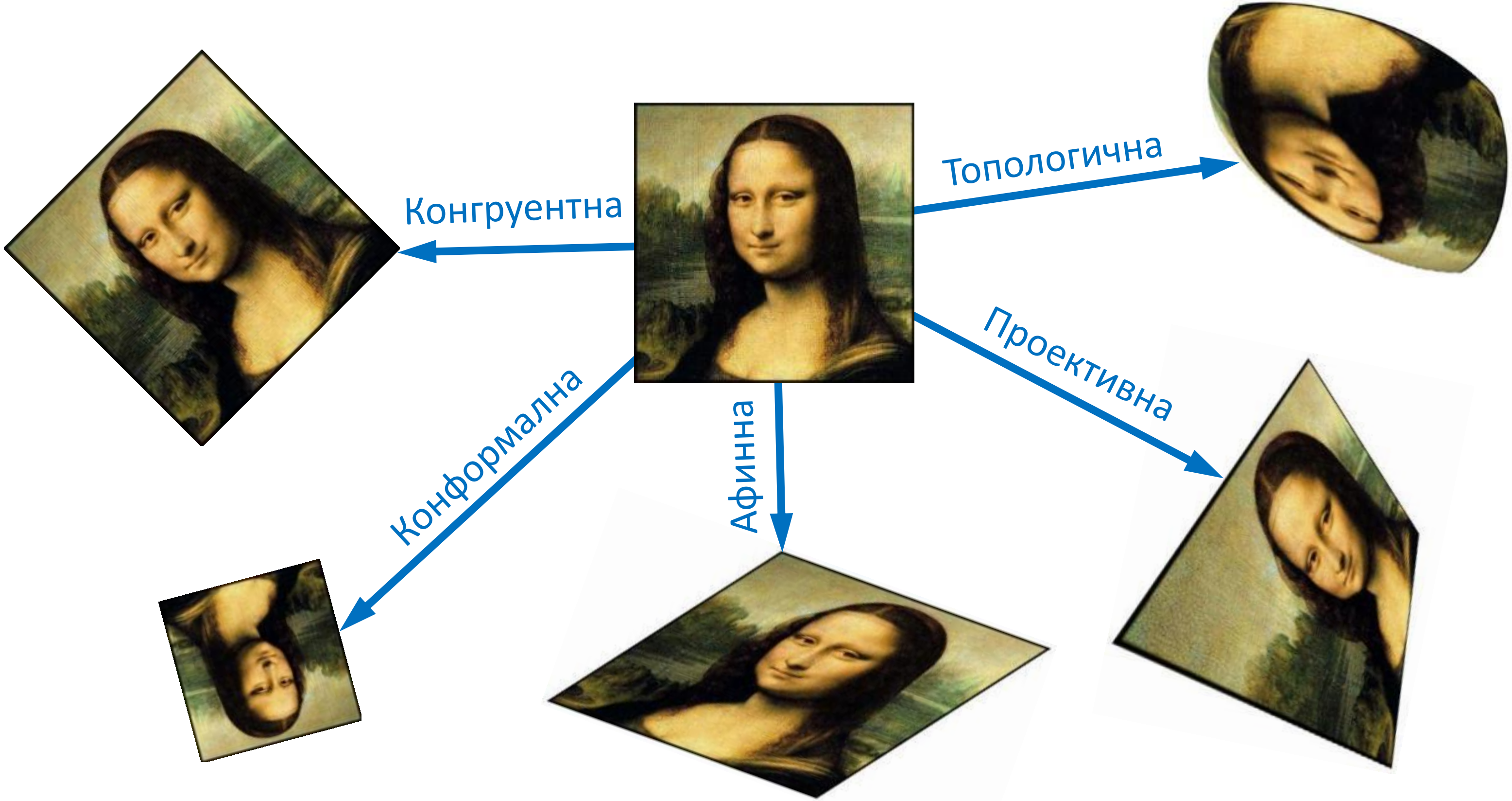
- Всяват страх у непросветените

Геометрии

Геометрии в Компютърната графика

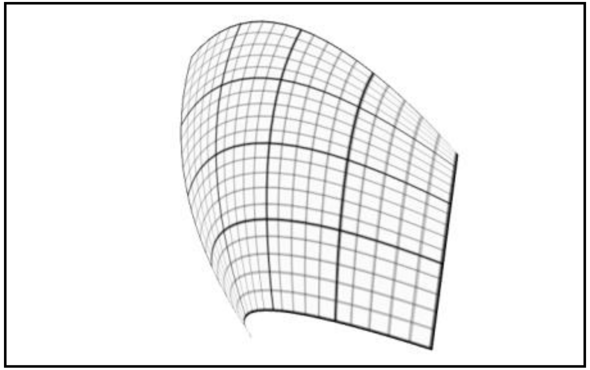
- Конгруентна геометрия
  (еднаквостна геометрия, напр. Евклидовата)
- Конформална геометрия
  (геометрия на подобностите)
- Афинна геометрия
- Проективна геометрия
- Топологична геометрия

Различните геометрии



Да ги видим

- Плочка се деформации според свойствата на различни геометрии



Сравнение

Сравнение на геометриите

– В някои геометрии само някои трансформации (т.е. матрици) са приложими

	Транслация и ротация	Промяна на мащаб	Скосяване	Централна проекция	Сферично отражение
Конгруентна	ДА	---	---	---	---
Конформална	ДА	ДА	---	---	---
Афинна	ДА	ДА	ДА	---	---
Проективна	ДА	ДА	ДА	ДА	---
Топологична*	ДА	ДА	ДА	ДА	ДА

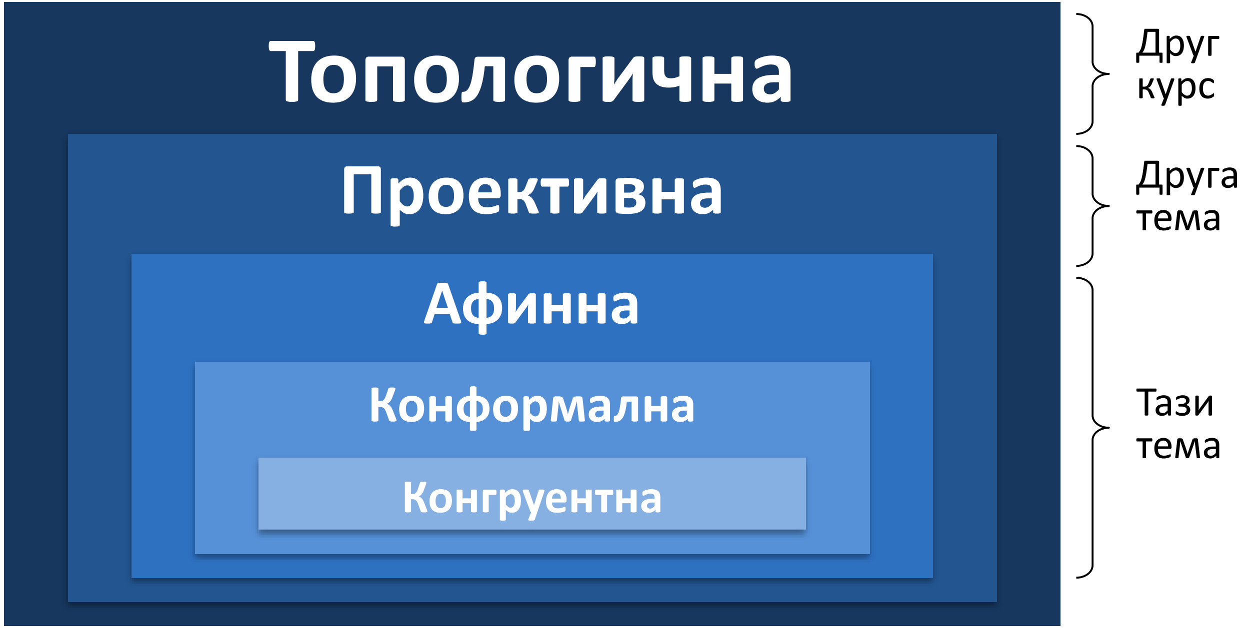
– Също и само някои свойства се запазват в някои геометрии

	Дължини	Ъгли	Успоредност	Линейност	Инцидентност
Конгруентна	ДА	ДА	ДА	ДА	ДА
Конформална	---	ДА	ДА	ДА	ДА
Афинна	---	---	ДА	ДА	ДА
Проективна	---	---	---	ДА	ДА
Топологична	---	---	---	---	ДА

Мощност

Друг факултет

Мощност на геометриите



Пример с трансформационна матрица

- Ако има коефициенти за скосяване
- Не е подходяща за афинни трансформации
- Използването ѝ ще развали ъглите
  (В компютърната графика „използване на матрица“ означава умножение на вектори или матрици с тази матрица)

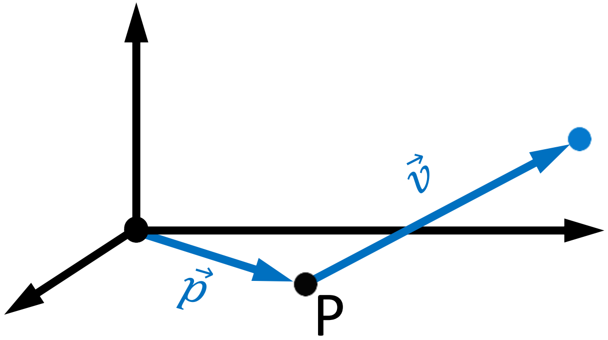
Трансляция

Транслация

Транслация без матрица

- Стандартно събиране на вектори

  $P + \vec{v} = \left(p_x + v_x, p_y + v_y, p_z + v_z\right)$



Транслация с матрица

Транслация по осите X, Y и Z

- Базисни матрици T_x, T_y и T_z при транслация на разстояние d

$$T_x(d) = \begin{pmatrix} 1 & 0 & 0 & d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, T_y(d) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, T_z(d) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Матрица T при транслация с вектор $\vec{v}$
  $T(\vec{v}) = T_x(v_x)T_y(v_y)T_z(v_z)$

- Да намерим явен вид на $T(\vec{v})$
- $T(\vec{v}) = T_x(v_x)T_y(v_y)T_z(v_z)$

$$= \begin{pmatrix} 1 & 0 & 0 & v_x \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & v_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & v_z \\ 0 & 0 & 0 & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} 1 & 0 & 0 & v_x \\ 0 & 1 & 0 & v_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & v_z \\ 0 & 0 & 0 & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} 1 & 0 & 0 & v_x \\ 0 & 1 & 0 & v_y \\ 0 & 0 & 1 & v_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \longleftarrow \vec{v}(v_x, vy, vz)$$

Пример

Пример с транслация

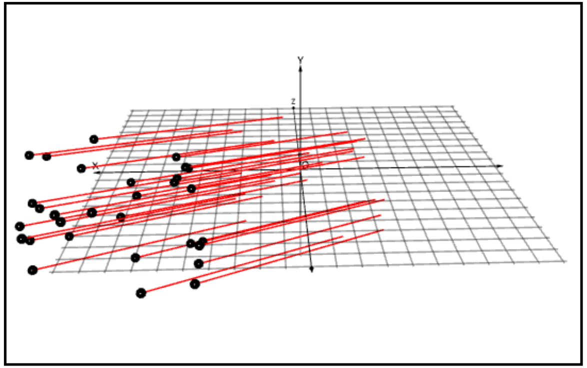
- Точка $(4,-2,0)$
- Преместваме с вектор $(2,3,-1)$

$$\begin{pmatrix} 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 \\ -2 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 6 \\ 1 \\ -1 \\ 1 \end{pmatrix} = (6,1,-1)$$

Реализация

Реализация на транслация по вектор

– Пример за транслация

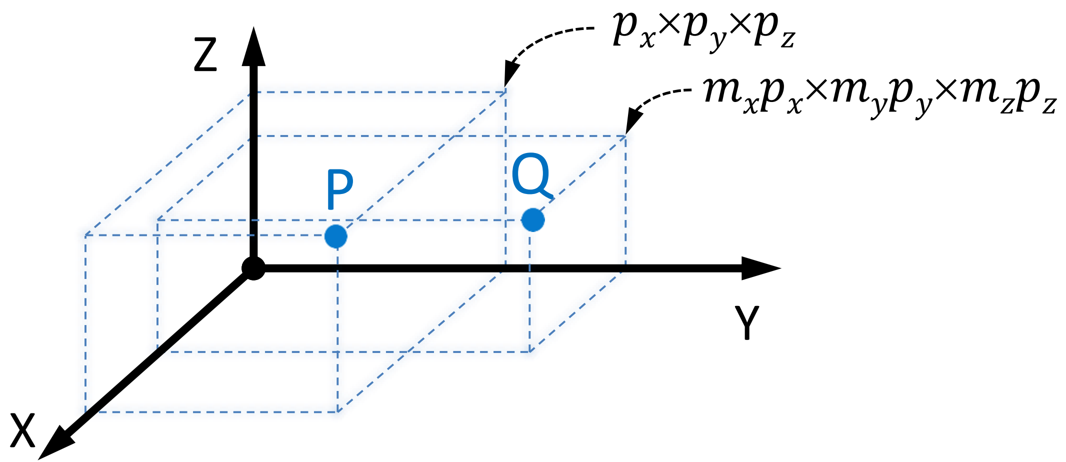


Мащабиране

Мащабиране

Мащабиране без матрица

- Приемаме в общия случай за 3D мащаб да се задава с вектор $\vec{m}(m_x, m_y, m_z)$



Реализация

- Чрез умножение на координатите

$$\left| \begin{aligned} q_x &= m_x p_x \\ q_y &= m_y p_y \\ q_z &= m_z p_z \end{aligned} \right.$$

- При конформалната геометрия мащабът трябва да е един: $m_x = m_y = m_z$
- При афинната може да е различен по о̀сите
- Бонус 3т: а при конгруентна геометрия?

Мащабиране с матрица

Мащабирания по осите X, Y и Z

- Базисни матрици S_x, S_y и S_z при мащаб с коефициент m

$$S_x(m) = \begin{pmatrix} m & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, S_y(m) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & m & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, S_z(m) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & m & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Матрица S при мащабиране с вектор $\vec{v}$
- $S(\vec{v}) = S_x(v_x)S_y(v_y)S_z(v_z)$

- Да намерим явен вид на $S(\vec{v})$
- $S(\vec{v}) = S_x(v_x)S_y(v_y)S_z(v_z)$

$$= \begin{pmatrix} v_x & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & v_y & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & v_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} v_x & 0 & 0 & 0 \\ 0 & v_y & 0 & 0 \\ 0 & 0 & v_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$\vec{v}(v_x, vy, vz)$

Пример

Пример с мащабиране

- Точка $(4, -2{,}5)$
- Мащабираме с вектор $(0.5{,}1.5{,}0.8)$

$$\begin{pmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 1.5 & 0 & 0 \\ 0 & 0 & 0.8 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 \\ -2 \\ 5 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ -3 \\ 4 \\ 1 \end{pmatrix} = (2, -3{,}4)$$

Конформалност

Конформалност в мащабирането

- Мащабите по осите са равни
- Ползват се и хомогенни координати

$$S(m)P = (mp_x, mp_y, mp_z) = (mp_x, mp_y, mp_z, 1) = \left(p_x, p_y, p_z, \frac{1}{m}\right)$$

$$S(m) = \begin{pmatrix} m & 0 & 0 & 0 \\ 0 & m & 0 & 0 \\ 0 & 0 & m & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{1}{m} \end{pmatrix}$$

Съставно мащабиране

Мащабиране спрямо 3D точка

– Базисното мащабиране е винаги спрямо $(0,0,0)$

Алгоритъм

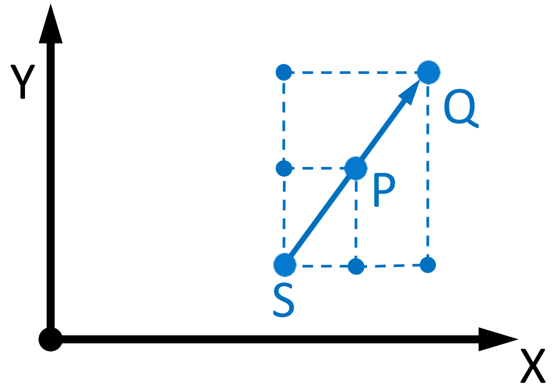
– Транслираме така, че точката спрямо която мащабираме да попадне в $(0,0,0)$

– Мащабираме

– Връщаме точката с обратна транслация

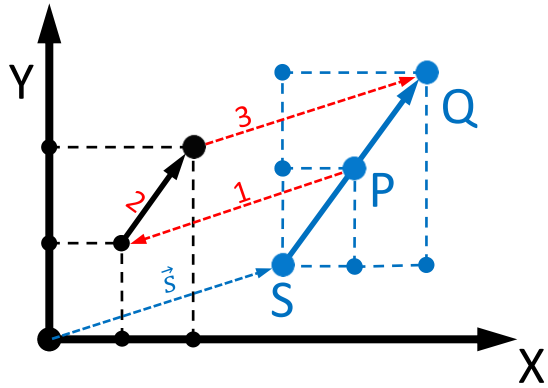
Илюстрация

Искаме



$$M = ?$$

Правим

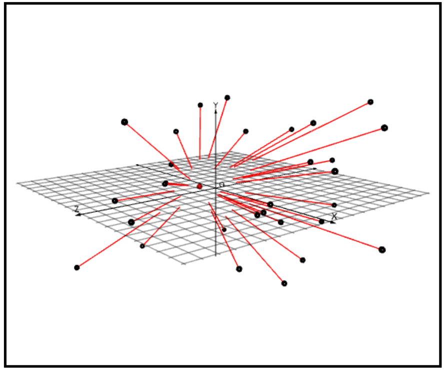


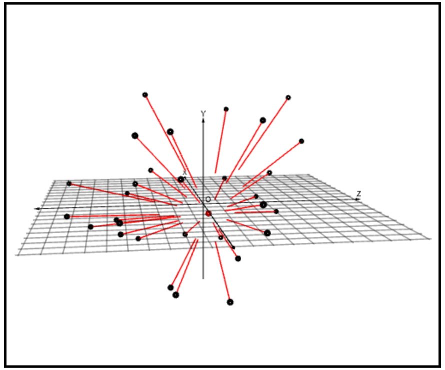
$$M = T(\vec{s})S(\vec{v})T(-\vec{s})$$

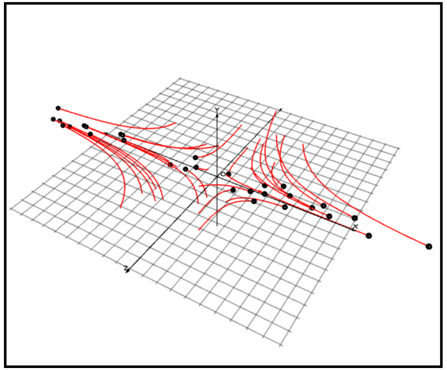
Реализация

Реализация на мащабиране

- Еднакъв мащаб
- С хомогенни координати
- С мащабиращ вектор







Ротация

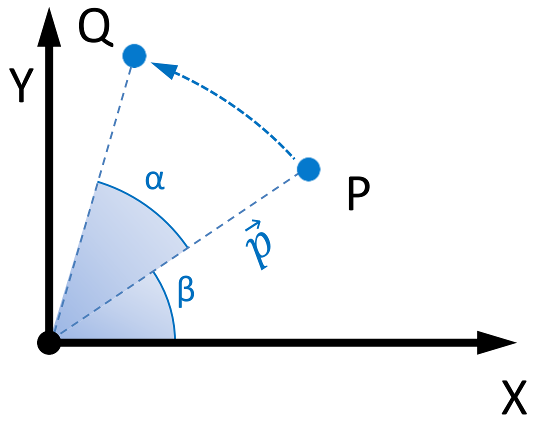
Ротация

Ротация без матрица

– Рядко се прилага в 3D, по-често в 2D

– От $|\vec{p}|$, α и β лесно намираме Q

$$\left|\begin{array}{l} q_x = p_x \cos\alpha - p_y \sin\alpha \\ q_y = p_x \sin\alpha + p_y \cos\alpha \end{array}\right.$$



С матрица

$$Q = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} P$$

Ротация с матрица

Базисни ротации в 3D

– Около оста Z
(това е същото въртене като в предходния 2D пример)

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

Като в 2D

– Проверяваме и получаваме очакваното
$R_z(\alpha)P = \left(p_x \cos\alpha - p_y \sin\alpha, p_x \sin\alpha + p_y \cos\alpha, p_z\right)$

Ротация около другите оси

- Получават се аналогично
- Ротация около X

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

Като в 2D

- Ротация около Y

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

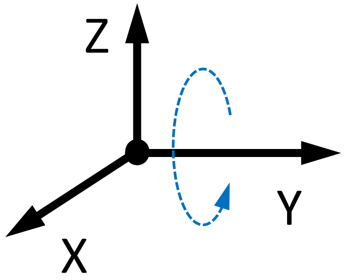
Почти като в 2D

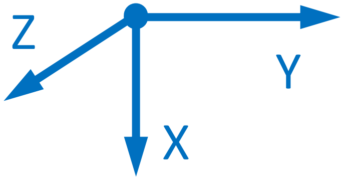
Защо?!?

- Да проверим къде отива $\vec{X}$ при ъгъл $\alpha = \frac{\pi}{2}$

$$R_y\left(\frac{\pi}{2}\right)\vec{X} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} = -\vec{Z}$$

- т.е. R_y върти „надолу“, от $\vec{Z}$ към $\vec{X}$





- При обратно въртене, от $\vec{X}$ към $\vec{Z}$ или $\alpha = -\frac{\pi}{2}$, матрицата ще е както очаквахме

Да проверим

- Матрица за въртене от $\vec{X}$ към $\vec{Z}$

$$\overline{R_y}(\alpha) = \begin{pmatrix} \cos\alpha & 0 & -\sin\alpha & 0 \\ 0 & 1 & 0 & 0 \\ \sin\alpha & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- А ето и при обратен ъгъл

- $R_y(-\alpha) = \begin{pmatrix} \cos\alpha & 0 & -\sin\alpha & 0 \\ 0 & 1 & 0 & 0 \\ \sin\alpha & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \overline{R_y}(\alpha)$

- Та ето затова!

$$\begin{array}{cc} & \begin{array}{cccc} \text{от} & \text{към} & \text{около} & \end{array} \\ \begin{array}{c} \text{от} \\ \text{към} \\ \text{около} \\ \end{array} & \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

$R_z(\alpha)$

$$\begin{array}{cc} & \begin{array}{cccc} X & Y & Z & \end{array} \\ \begin{array}{c} X \\ Y \\ Z \\ \end{array} & \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

$R_y(\alpha)$

$$\begin{array}{cc} & \begin{array}{cccc} Z & X & Y & \end{array} \\ \begin{array}{c} Z \\ X \\ Y \\ \end{array} & \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{array} \xrightarrow[\text{в реда XYZ}]{\text{подреждаме}} \begin{array}{cc} & \begin{array}{cccc} X & Y & Z & \end{array} \\ \begin{array}{c} X \\ Y \\ Z \\ \end{array} & \begin{pmatrix} \cos\alpha & 0 & \sin\alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\alpha & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

ВАЖНО

Ако в някой източник намерите матриците по по-различен начин, проверете следните неща:

- Дали са за лява или за дясна координатна система
- Дали се умножават пред или зад векторите
- Дали редът на осите е XYZ
- Дали няма печатна грешка

Обобщена матрица

Не става в общия случай

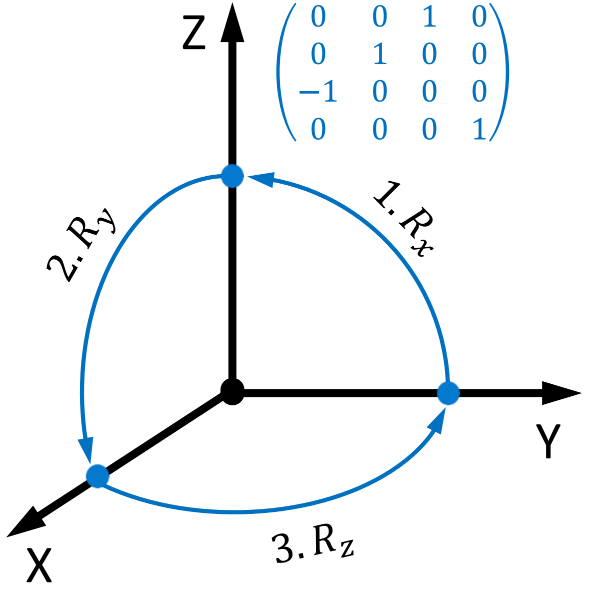
– При транслации редът на прилагане на T_x, T_y и T_z може да е произволен
– При ротации редът на прилагане на R_x, R_y и R_z **не може** да е произволен

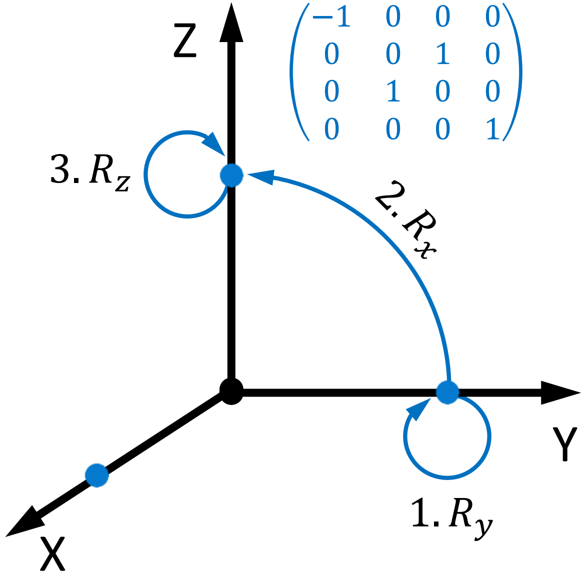
Пример с въртене на 90 градуса

– Вариант 1: около $\vec{X}$, около $\vec{Y}$, около $\vec{Z}$
– Вариант 2: около $\vec{Y}$, около $\vec{X}$, около $\vec{Z}$

Точка $(0,1,0)$

- При вариант 1: $(0,1,0) \rightarrow (0,1,0)$
- При вариант 2: $(0,1,0) \rightarrow (0,0,1)$





ПАК ВАЖНО

Ако прилагаме първо матрица M_1, после M_2 и накрая M_3:

- Общата матрица е $M = M_3 M_2 M_1$
- Редът е обратен – първата приложена трансформация записваме последна
- Помни се, като че ли M_i са функции: $MA = M_3 M_2 M_1 A = M_3\left(M_2\left(M_1(A)\right)\right)$

Съставна ротация

Ротация в 2D около точка

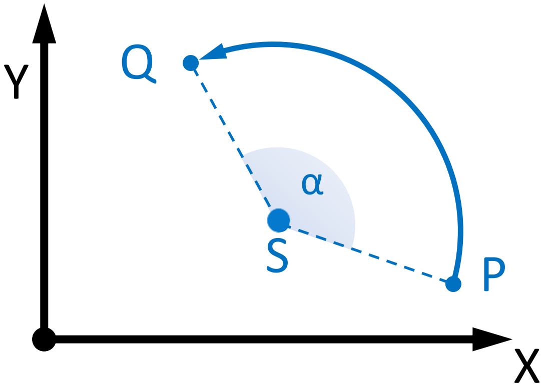
- Базисната ротация в 2D е винаги около точката $(0,0)$

Алгоритъм

- Транслираме така, че точката около която въртим да попадне в $(0,0)$
- Въртим около $(0,0)$
- Връщаме точката с обратна транслация

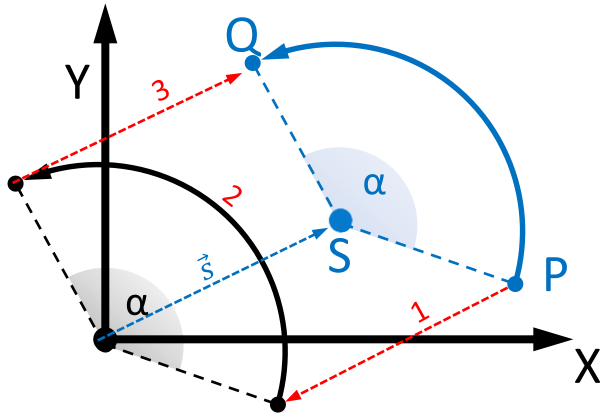
Илюстрация

Искаме



$$M = ?$$

Правим

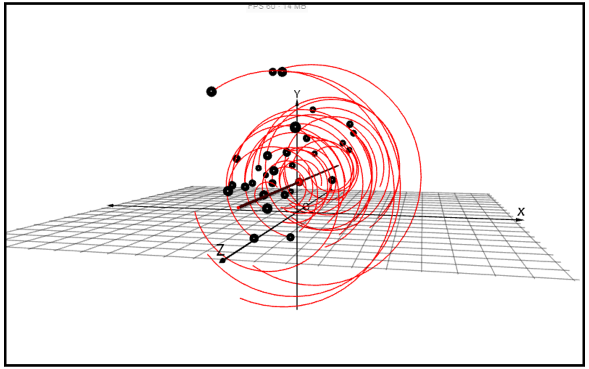


$$M = T(\vec{s})R_z(\alpha)T(-\vec{s})$$

Реализация

Реализация на ротация около точка

– Пример за 2D ротация ... в 3D

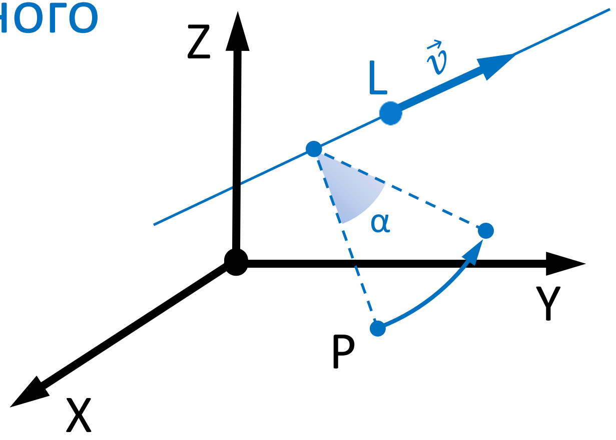


Снимка: Щефан Биневийс, http://www.capella-observatory.com

Въртене около права

Дадена е права от точка L и вектор $\vec{v}$

- Въртим друга точка P около тази права
- Използваме само базисни матрици, но за сметка на това са много



Алгоритъм

Алгоритъм на въртене

1. Транслираме $L \rightarrow$ точката $(0,0,0)$
2. Въртим около Z, че $\vec{v} \rightarrow$ равнината XZ
3. Въртим около Y, че $\vec{v} \rightarrow$ оста Z
4. Въртим P около Z на желания ъгъл
5. Правим обратното на 3
6. Правим обратното на 2
7. Правим обратното на 1

Матрици M_1 и M_7

Първата и последната матрици – транслации

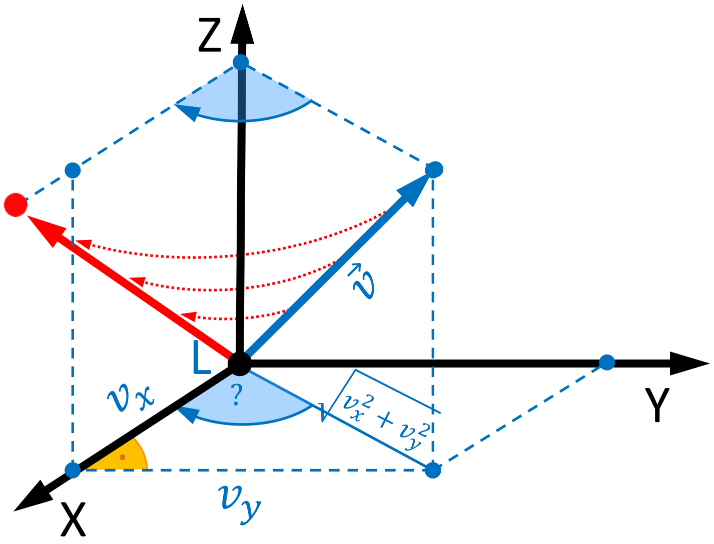
- Ако $\vec{p} = \overrightarrow{OP}$, то P се транслира в O с $T(-\vec{p})$
- Обратната транслация е $T(\vec{p})$

$$M_1 = T(-\vec{p}) = \begin{pmatrix} 1 & 0 & 0 & -p_x \\ 0 & 1 & 0 & -p_y \\ 0 & 0 & 1 & -p_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad M_7 = T(\vec{p}) = \begin{pmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Матрици M_2 и M_6

Въртене без да знаем ъгъла

– Въртим $\vec{v}$ около Z, за да попадне в равнината XZ



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

$$\cos ? = \frac{v_x}{\sqrt{v_x^2+v_y^2}} \quad \sin ? = \frac{v_y}{\sqrt{v_x^2+v_y^2}}$$

- Така за матрица M_2 получаваме

$$M_2 = \begin{pmatrix} \frac{v_x}{\sqrt{v_x^2+v_y^2}} & \frac{v_y}{\sqrt{v_x^2+v_y^2}} & 0 & 0 \\ \frac{-v_y}{\sqrt{v_x^2+v_y^2}} & \frac{v_x}{\sqrt{v_x^2+v_y^2}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Разликата е само в тези минуси

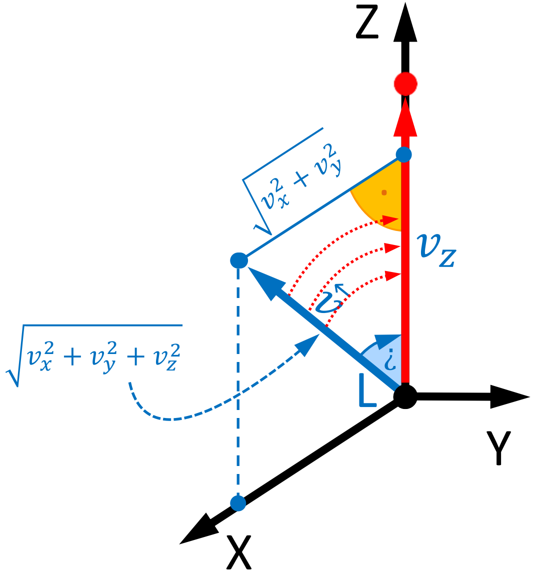
- За M_6 въртенето е в обратна посока

$$M_6 = \begin{pmatrix} \frac{v_x}{\sqrt{v_x^2+v_y^2}} & \frac{-v_y}{\sqrt{v_x^2+v_y^2}} & 0 & 0 \\ \frac{v_y}{\sqrt{v_x^2+v_y^2}} & \frac{v_x}{\sqrt{v_x^2+v_y^2}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Матрици M_3 и M_5

Аналогично на M_2 и M_6

- Завъртаме $\vec{v}$ около Y, за да попадне върху оста Z



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

$$\cos ¿ = \frac{v_x}{\sqrt{v_x^2 + v_y^2 + v_z^2}} \qquad \sin ¿ = \frac{\sqrt{v_x^2 + v_y^2}}{\sqrt{v_x^2 + v_y^2 + v_z^2}}$$

- Така за матрица M_3 и M_5 получаваме

$$M_3 = \begin{pmatrix} \frac{v_z}{\sqrt{v_x^2+v_y^2+v_z^2}} & 0 & \frac{-\sqrt{v_x^2+v_y^2}}{\sqrt{v_x^2+v_y^2+v_z^2}} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{\sqrt{v_x^2+v_y^2}}{\sqrt{v_x^2+v_y^2+v_z^2}} & 0 & \frac{v_z}{\sqrt{v_x^2+v_y^2+v_z^2}} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$M_5 = \begin{pmatrix} \frac{v_z}{\sqrt{v_x^2+v_y^2+v_z^2}} & 0 & \frac{\sqrt{v_x^2+v_y^2}}{\sqrt{v_x^2+v_y^2+v_z^2}} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{-\sqrt{v_x^2+v_y^2}}{\sqrt{v_x^2+v_y^2+v_z^2}} & 0 & \frac{v_z}{\sqrt{v_x^2+v_y^2+v_z^2}} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Матрица M4

Остава матрица M_4

- Въртене около Z на желания ъгъл α

$$M_4 = R_z(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Цялостната трансформация е $M = M_7 M_6 M_5 M_4 M_3 M_2 M_1$
- Можем да ги умножим ръчно и да видим как изглежда M, но резултатът е … франкенщайново-квазимодов

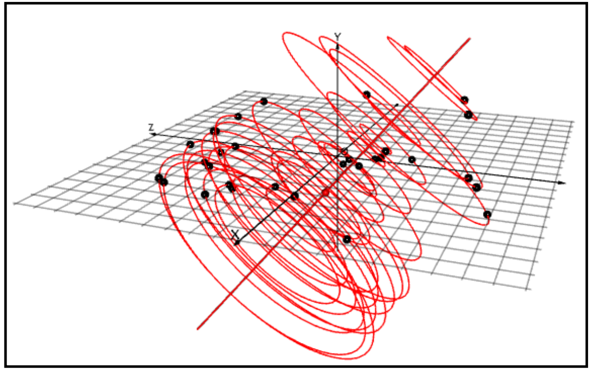
Затова

- Нека оставим умножението на софтуера
- Често може да се намери пълното изписване на M, но за частен случай, като например за единичен вектор $\vec{v}$

Реализация

Реализация на ротация около линия

– Пример за ротация в 3D



Въпроси?

Повече информация

[AGO1]	стр. 67-71	[LENG]	стр. 71-80
[ALZH]	гл. 3	[MORT]	стр. 47-68
[BAGL]	стр. 135-136	[PARE]	стр. 39-42
[GRIM]	стр. A11-A22	[SEAK]	стр. 31-33
[KLAW]	стр. 100-104	[VINC]	стр. 51-73
[KLRO]	стр. 13-15	[ZHDA]	стр. 209-224

А също и:

– Rotation About an Arbitrary Axis in 3 Dimensions
http://inside.mines.edu/~gmurray/ArbitraryAxisRotation/

Край