Animación por Ordenador

Tema 4

La animación como cambio.
Fotogramas clave. Intercalado.
Funciones de movimiento. Orientación.

CONTENIDO

LA ANIMACIÓN COMO CAMBIO

LA ANIMACIÓN COMO CAMBIO

- Cuando hablamos de ‘animación’ nos referimos a definir la información necesaria para ‘posar’ o cambiar un elemento de nuestra escena a lo largo de un determinado periodo de tiempo.
- Esto incluye la información necesaria para especificar todos los valores de los grados de libertad (DOF) que aparecen en la escena a lo largo del tiempo.
- Degree of Freedom (DOF): Una variable φ que describa un valor de transformación o cambio de un parámetro en un elemento de la escena.
- Cambiando los grados de libertad en el tiempo nos dará la animación resultante.

ESPACIO DE POSES

Si un objeto tiene N DOFs, entonces una 'pose' puede verse como un punto N-dimensional en el *espacio de poses*

$$\mathbf{\Phi} = \begin{bmatrix} \phi_1 & \phi_2 & \dots & \phi_N \end{bmatrix}$$

Una animación puede verse como un 'movimiento' de un punto en el *espacio de poses, o* como una curva en dicho espacio en función del tiempo:

$$\mathbf{\Phi} = \mathbf{\Phi}(t)$$

Generalmente, pensamos en animaciones individuales como una curva continua, pero puede haber discontinuidades, lo que producirá 'cortes' en la animación.

CANALES

Si una animación es una curva N-dimensional en el *espacio de poses*, podemos separarla en N curvas de una dimensión cada una para un DOF

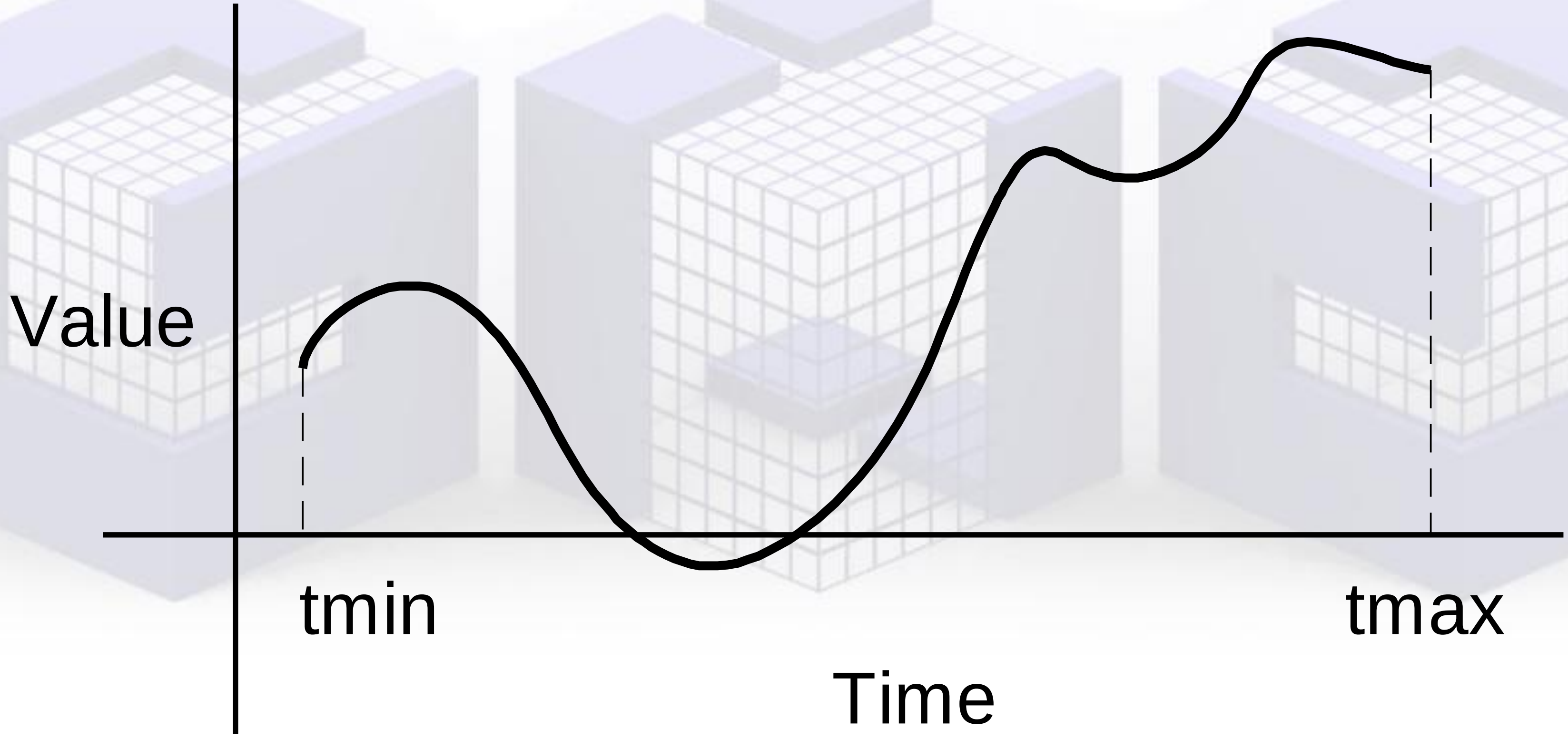
$$\phi_i = \phi_i(t)$$

Es lo que podemos llamar '***canales***' de nuestra animación.

Un *'canal'* almacena el valor de una función 1D sobre un determinado dominio.

Un canal contiene la animación de un grado de libertad.

CANALES



CANALES

- Como un *canal* representa los datos almacenados para el cambio de valor de un DOF, evaluando el canal para un determinado valor de *t* siempre devolverá el mismo resultado.
- Podemos encontrarnos canales con discontinuidad en valores, pero no en el tiempo (debemos ser capaces de evaluar el canal en cuanquier instante).
- La mayoría de las veces un canal se usa para representar cambios de un DOF en el tiempo, pero podemos usarlo también para relacionar otras variables (i.e., color, iluminación, …)

VECTOR DE CANALES

- Una animación puede almacenarse como una conjunto de canales (un vector, por ejemplo)
- Podemos entender ese vector de canales como un almacenamiento de un vector de muestras regularmente separadas en el tiempo.
- Con esta idea, podemos almacenar una animación en un array 2D de flotantes (N_DOFs x N_Frames)
- Pero también podemos almacenarlo como un vector de canales, donde cada canal es responsible de almacenar los datos sobre un DOF.

VECTOR DE POSES

- Un modo alternativo de almacenar una animación es como un conjunto de poses (un vector)
- Pdemos por tanto almacenarlo como un array 2D de flotantes (N_Frames x N_DOFs)

Pero, qué es mejor, almacenar *poses* o *canales?*

POSES VS. CANALES

- Qué es mejor?
  - Depende de los requisitos que tengamos.

- Debemos tener en cuenta que:
  - Almacenar y gestionar *poses* es más rápido
  - Los *canales* nos dan más flexibilidad y normalmente necesitan menos memoria

VECTOR DE POSES

- El método de almacenamiento de animaciones mediante vector de poses es la forma más rápida de reproducir animaciones.
- Una pose (vector de flotantes) es lo que el ordenador necesita para 'colocar' una objeto en la escena, pues define sus parámetros en un momento concreto.
- La información estará almacenada de forma contigua, por lo que podrá ser accedida a través de su dirección directa.

VECTOR DE CANALES

- Como cada canal está almacenado independientemente, podemos aprovecharlo para usar diferentes opciones de almacenamiento y maximizar eficiencia en el almacenamiento.
- Además, podemos crear y modificatr nuevos canales sin problema (muy complicado usando poses, pues deberíamos modificar toda la estructura).
- Sin embargo, necesitamos evaluar de forma independiente cada canal para acceder a toda la información de la escena en un instante de tiempo concreto (una *pose*), lo que implica tiempo y aceso discontinuo a la memoria.

POSES VS. CHANNELS

- Un *vector de poses* es mejor si necesitamos reproducir animaciones sencillas y máxima ***eficiencia***, como ocurre por ejemplo en videojuegos.
- *Vector de canales* es mejor si queremos mayor ***flexibilidad*** en el sistema de animación, como ocurre en las producciones para cine y tv.
- Los *vectores de canales* son útiles en juegos más sofisticados o en casos donde el almacenamiento es más crítico que el rendimiento de la CPU.

CONTINUIDAD TEMPORAL

- Algunas veces pensamos en las animaciones para reproducirla a una determinada velocidad (framerate), por ejemplo 25fps
- Pero a menudo es mejor pensar en la animación como una continuidad en el tiempo, y no a una determinada velocidad de cuadros, por ejemplo:
  - Para conversión de formatos (Film / NTSC / PAL)
  - Manipulación en tiempo real.
  - Cálculo de *Motion blur*
- Ciertos efectos y movimientos rápidos pueden necesitar poder acceder realmente a todos los fotrogramas de una animación de forma individual.

ALMACENAMIENTO DE ANIMACIONES

- Independientemente de si pensamos en una animación como un proceso continuo o como un conjunto de puntos discretos, debemos considerar qué métodos de almacenamiento de datos usamos.
- Algunos de estos métodos pueden requerir algún tipo de discretización temporal, mientras que otros no.
- Incluso cuando almacenamos un canal en incrementos de tiempo (claves), es mejor pensar en ello como una función continua interpolando el tiempo entre valores.

FOTOGRAMAS CLAVE

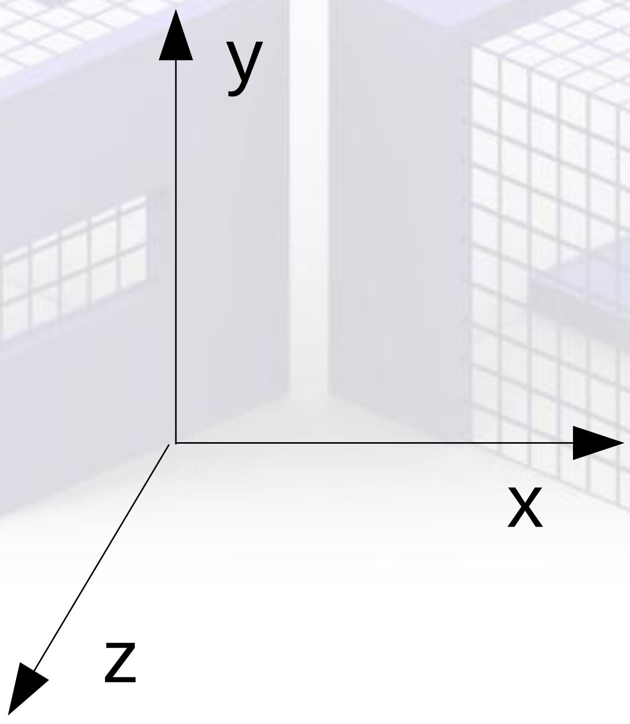
FOTOGRAMAS CLAVE O KEYFRAMES

- Los *fotogramas clave*, o *Keyframes,* son una alternativa para almacenar datos de canales de animación.
- En vez de almacenar todos los valores en el tiempo de los valores de un canal, solo almacenamos algunos en determinados instantes de tiempo del rango almacenado en el canal.
- Necesitamos entonces conocer cómo se obtienen los valores que no están almacenados en el canal. Se hace mediante interpolación, existiendo diferentes tipos, como veremos más adelante.

UN REPASO AL ÁLGEBRA LINEAL

SISTEMAS DE COORDENADAS

- Normalmente utilizamos sistemas de coordenadas siguiendo la *regla de la mano derecha*



VECTORES

$$\mathbf{a} = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix}$$

$$\mathbf{a} + \mathbf{b} = \begin{bmatrix} a_x + b_x & a_y + b_y & a_z + b_z \end{bmatrix}$$

$$\mathbf{a} - \mathbf{b} = \begin{bmatrix} a_x - b_x & a_y - b_y & a_z - b_z \end{bmatrix}$$

$$-\mathbf{a} = \begin{bmatrix} -a_x & -a_y & -a_z \end{bmatrix}$$

$$s * \mathbf{a} = \begin{bmatrix} s * a_x & s * a_y & s * a_z \end{bmatrix}$$

MAGNITUD DE UN VECTOR (MÓDULO)

La magnitud o módulo (longitud) de un vector es:

$$|\mathbf{v}| = \sqrt{v_x^2 + v_y^2 + v_z^2}$$

Vector unitario (módulo=1.0)

$$\frac{\mathbf{v}}{|\mathbf{v}|}$$

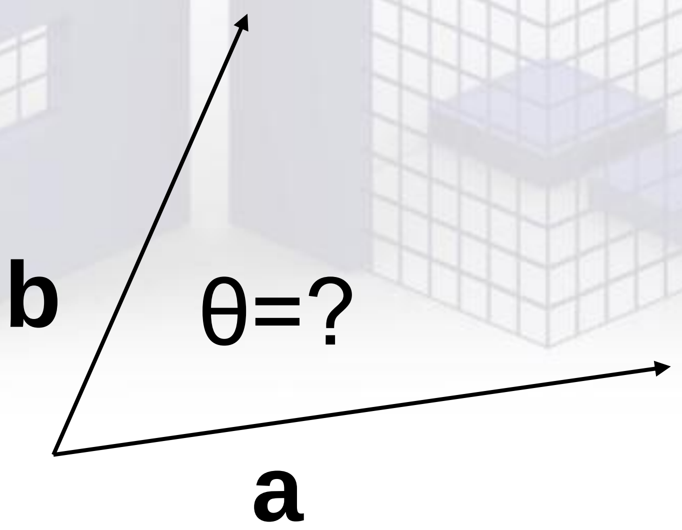
PRODUCTO ESCALAR

$$\mathbf{a}\cdot\mathbf{b} = a_x b_x + a_y b_y + a_z b_z = \sum a_i b_i$$

$$\mathbf{a}\cdot\mathbf{b} = |\mathbf{a}||\mathbf{b}|\cos\theta$$

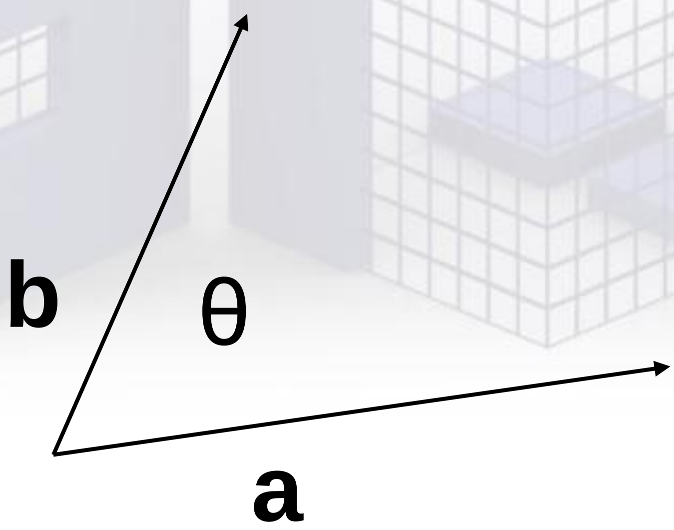
EJEMPLO: ÁNGULO ENTRE VECTORES

Cómo podemos encontrar el ángulo θ que forman dos vectores A y B?

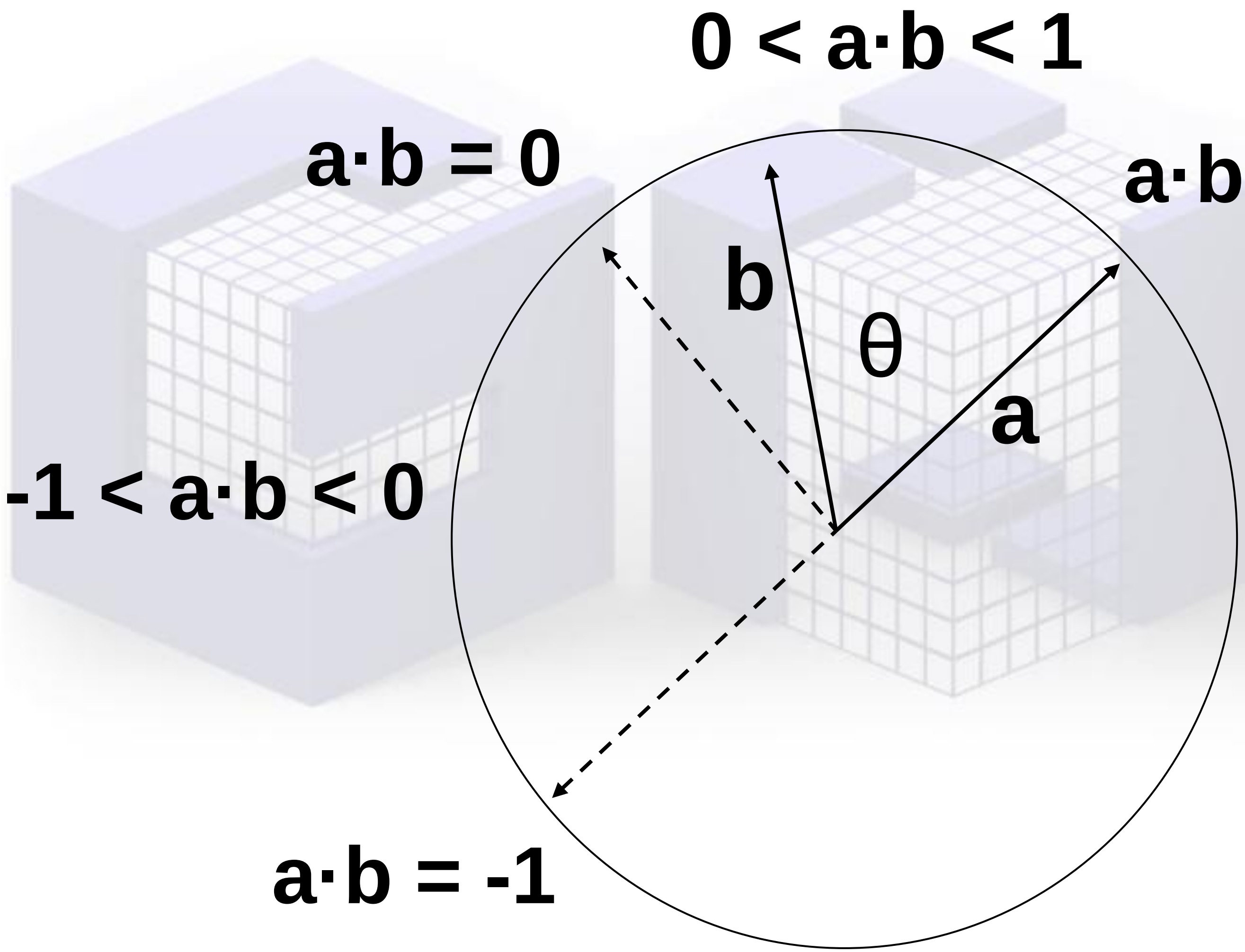


EJEMPLO: ÁNGULO ENTRE VECTORES

$$\theta = \cos^{-1}\left(\frac{\mathbf{a}\cdot\mathbf{b}}{|\mathbf{a}||\mathbf{b}|}\right)$$



PRODUCTOS ESCALARES CON VECTORES UNITARIOS



$$|\mathbf{a}| = |\mathbf{b}| = 1.0$$

$$\mathbf{a} \cdot \mathbf{b} = \cos(\theta)$$

PRODUCTOS ESCALARES CON VECTORES NO UNITARIOS

Si **a** y **b** son vectores arbitrarios (no unitarios), entonces las siguientes relaciones siguen siendo ciertas:

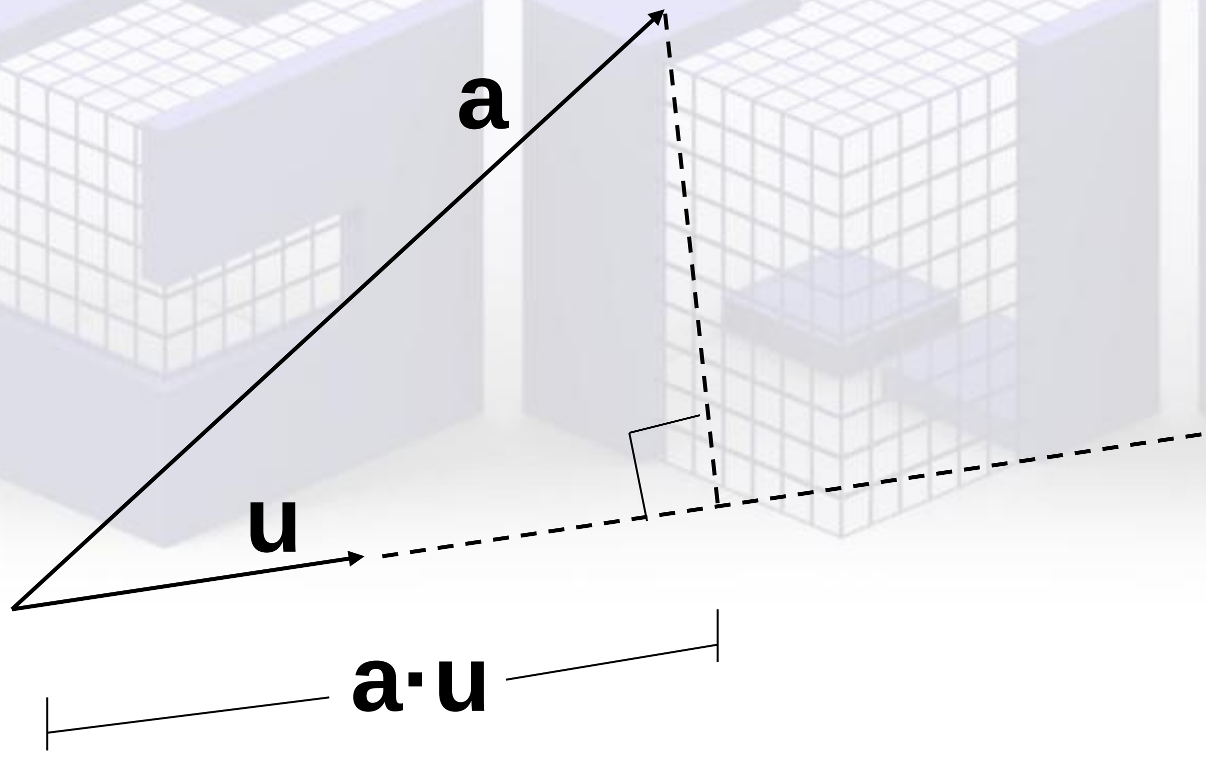
If θ < 90º then **a·b** > 0

If θ = 90º then **a·b** = 0

If θ > 90º then **a·b** < 0

PRODUCTOS ESCALARES CON UN VECTOR UNITARIO

Si **|u|=1.0** entonces **a·u** es la longitud de la *proyección* de **a** sobre **u**



PRODUCTO VECTORIAL

$$\mathbf{a}\times\mathbf{b} = \begin{vmatrix} i & j & k \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{vmatrix}$$

$$\mathbf{a}\times\mathbf{b} = \begin{bmatrix} a_y b_z - a_z b_y & a_z b_x - a_x b_z & a_x b_y - a_y b_x \end{bmatrix}$$

PROPIEDADES DEL PRODUCTO VECTORIAL

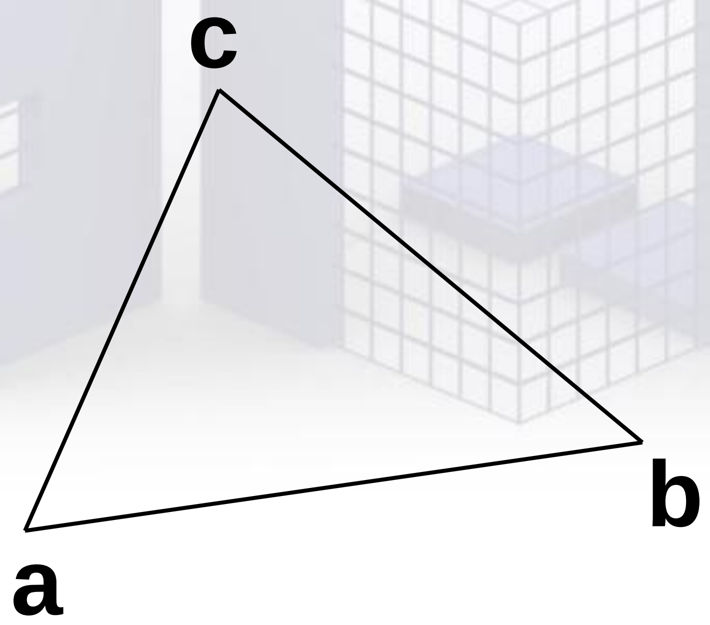
$$\left|\mathbf{a}\times\mathbf{b}\right| = \left|\mathbf{a}\right|\left|\mathbf{b}\right|\sin\theta$$

$\left|\mathbf{a}\times\mathbf{b}\right| =$ área del paralelogramo **ab**

$\mathbf{a}\times\mathbf{b}$ es perpendicular a los vectores **a** y **b**, con la dirección definida por la regla de la mano derecha. Se utiliza para el cálculo de vectores normales a las caras.

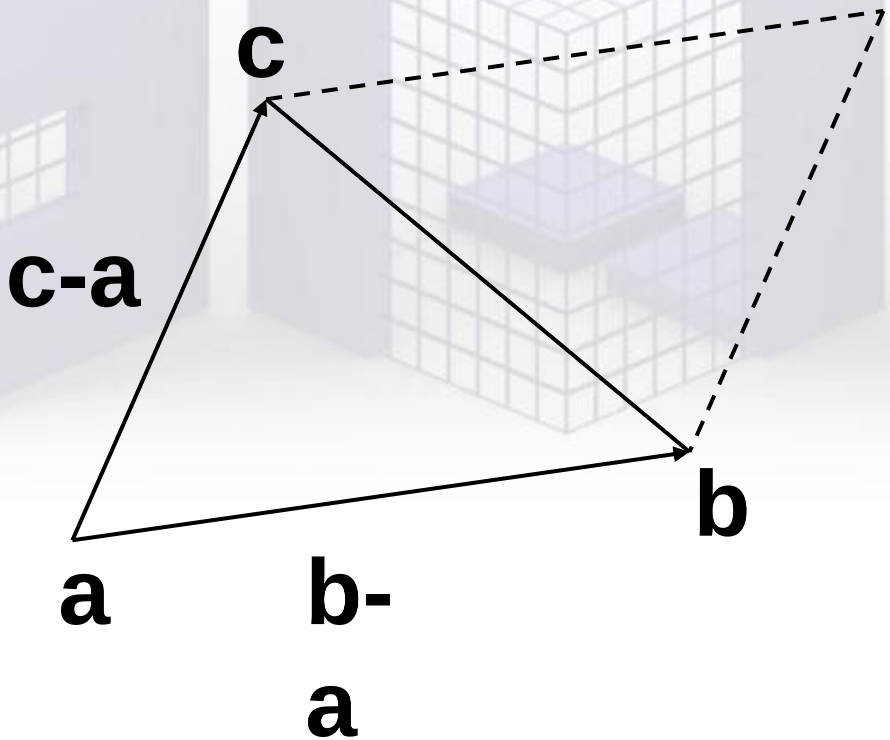
EJEMPLO: ÁREA DE UN TRIÁNGULO

Encontrar el área de un triángulo definido por 3 puntos 3D **a**, **b**, and **c**



EJEMPLO: ÁREA DE UN TRIÁNGULO

$$area = \frac{1}{2}\left|\left(\mathbf{b}-\mathbf{a}\right)\times\left(\mathbf{c}-\mathbf{a}\right)\right|$$

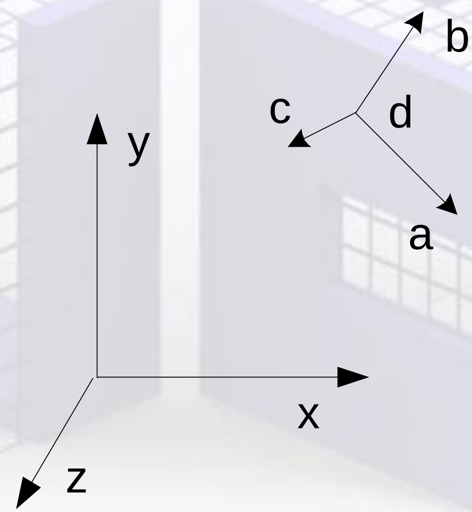


MATRICES

En gráficos por ordenador se utilizan normalmente matrices 4x4 con coordenadas homogeneas para definir transformaciones.

Una matriz de transformación rígida 4x4 tendrá la forma:

$$\mathbf{M} = \begin{bmatrix} a_x & a_y & a_z & 0 \\ b_x & b_y & b_z & 0 \\ c_x & c_y & c_z & 0 \\ d_x & d_y & d_z & 1 \end{bmatrix}$$



Donde **a, b**, y **c** son vectores unitarios ortogonales que representan la orientación, y **d** es un vector que representa posición.

TRANSFORMACIONES

Para transformar un vector **v** por la matríz **M**:

$v'=v\cdot M$

If aplicamos varias transformaciones, podemos multiplicar varias matrices:

$v'=(((v\cdot M_1)\cdot M_2)\cdot M_3)\cdot M_4\ldots$

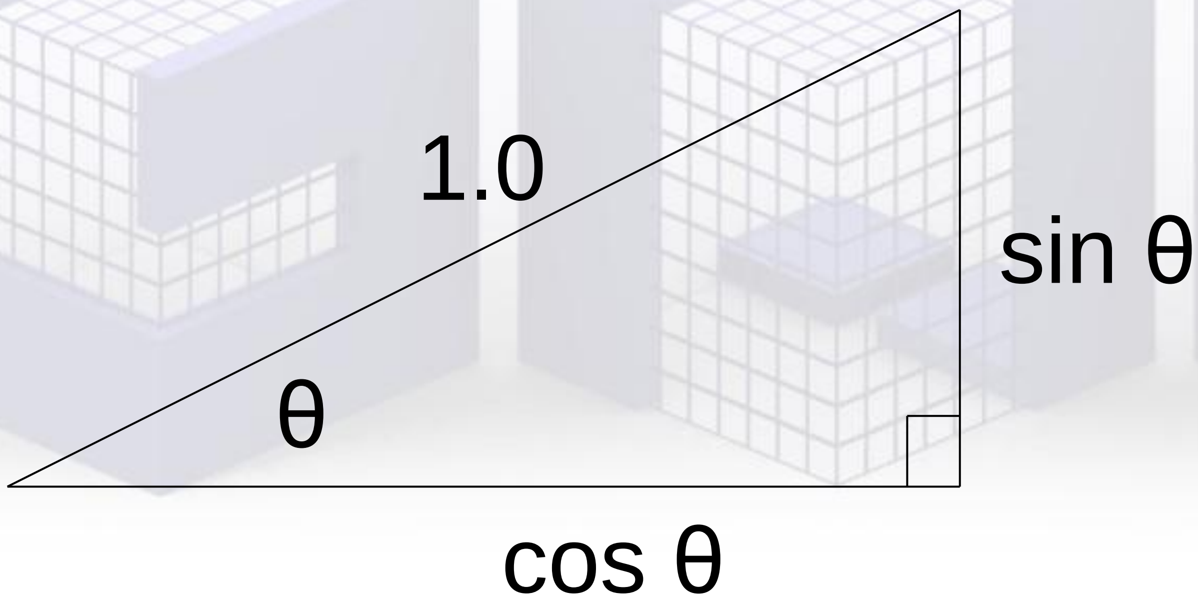
O podemos concatenar las transformaciones en una única matriz:

$M_{total}=M_1\cdot M_2\cdot M_3\cdot M_4\ldots$

$v'=v\cdot M_{total}$

TRIGONOMETRÍA

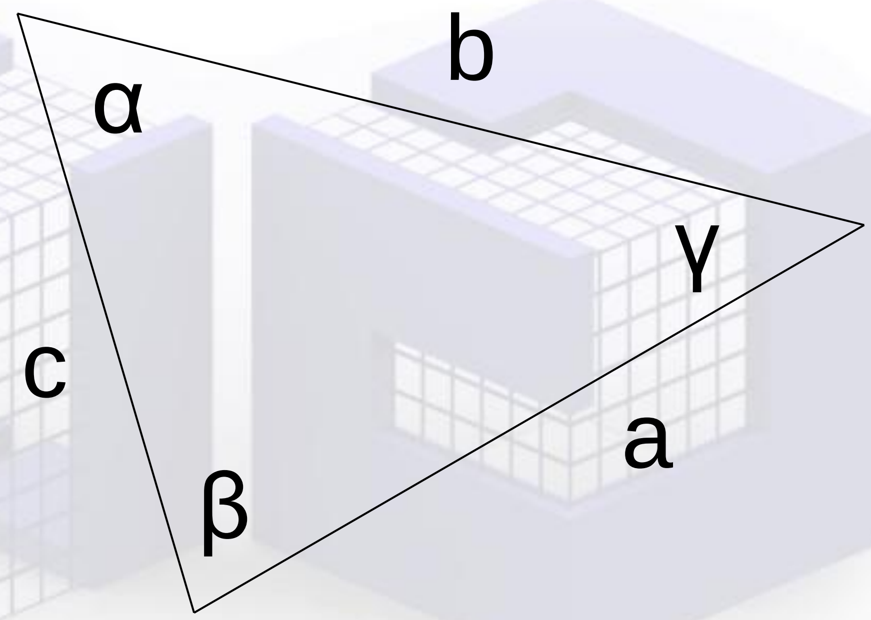
$$\cos^2\theta + \sin^2\theta = 1$$



LEYES DE LOS SENOS Y COSENOS

Ley de los senos:

$$\frac{a}{\sin \alpha} = \frac{b}{\sin \beta} = \frac{c}{\sin \gamma}$$



Ley de los cosenos:

$$c^2 = a^2 + b^2 - 2ab\cos\gamma$$