COMP3271 Computer Graphics

Orientation Representation

2019-20

Objectives

Focus on the rotation transformation

Four orientation formats

- Rotation matrices
- Euler angles
- Axis-angle representation
- Quaternions

Comparisons of these representations

Criteria for Orientation Formats

How much storage is needed for the representation?

- How many numbers are needed to represent an orientation/rotation?

How efficient to form new orientations? $R_2 \cdot R_1$

How efficient to rotate points and vectors? $R \cdot v$

How well the representation can be interpolated?

How suitable for numeric integration (e.g. for physical simulation)?

Rotation Matrices

$$R = \begin{pmatrix} u_0 & v_0 & w_0 \\ u_1 & v_1 & w_1 \\ u_2 & v_2 & w_2 \end{pmatrix}$$

The column vectors
$u = (u_0, u_1, u_2)^T, v = (v_0, v_1, v_2)^T,$
$w = (w_0, w_1, w_2)^T \in \mathbb{R}^3$
are three orthonormal basis vectors.

27M

Nine numbers needed for a rotation

- Euler's rotation theorem states that we just need three numbers to represent a rotation

$R_2 \cdot R_1$

New rotations are obtained by matrix-matrix multiplication; vectors are rotated by matrix-vector multiplication

$9M \rightarrow R \cdot V$

- Can be performed quite efficiently, some hardware has built-in circuitry for the multiplications

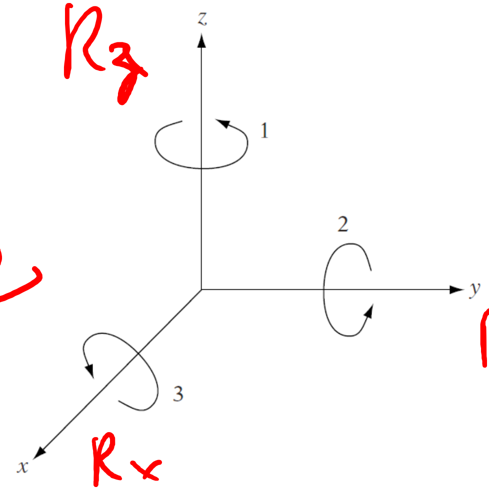
Euler Angles

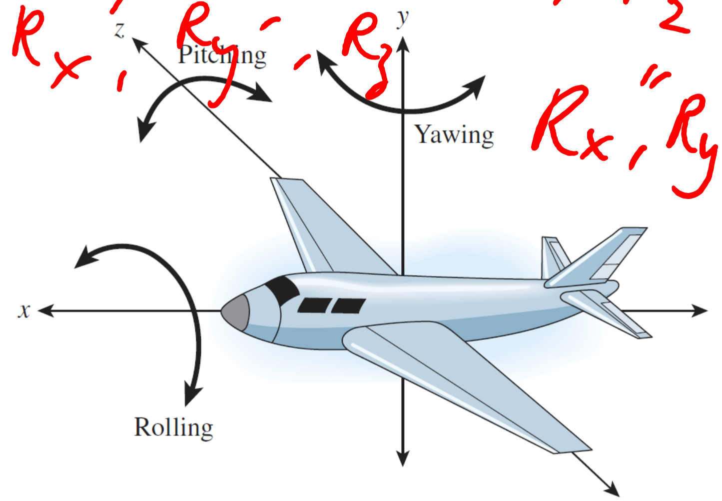
$R_1 : R_x, R_y, R_z$

$R_2 : R_x', R_y', R_z' \rightarrow R_2 \circ R_1$

R_x'', R_y'', R_z''

$\underbrace{R_x \times R_y \times R_z}$





Use 3 sequential rotations about a set of orthogonal axes to specify an orientation.

- If axes are fixed, need only 3 numbers for the angles (the Euler angles)
- If we choose the standard x-,y-,z-axes, the rotations are given by R_x, R_y, R_z
- No standard order for the use of the three axes

Composition of rotations and vector rotations resort to converting back to matrix representation and therefore are not efficient

Axis-Angle Representation

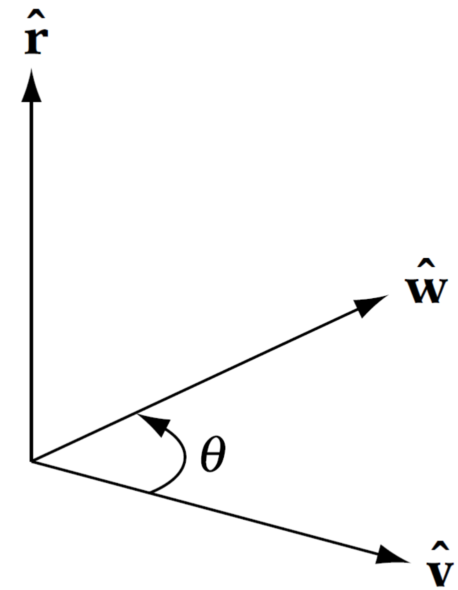
(x, y, z)

Represent a rotation by an axis of rotation $\mathbf{r}$, and the angle of rotation θ about this axis

- $\mathbf{r}$ is normalized so the degree of freedom is 3

The axis-angle rotation to bring a vector $\mathbf{v}$ to another vector $\mathbf{w}$ is given by

$$\mathbf{r} = \hat{\mathbf{v}} \times \hat{\mathbf{w}}$$
$$\theta = \arccos(\hat{\mathbf{v}} \cdot \hat{\mathbf{w}})$$



Composition of rotations and vector rotations are not trivial.

Quaternions

Mathematical object developed by Sir William Rowan Hamilton in 1843 as an extension to the complex numbers

General form of a quaternion:

$$\mathbf{q} = w + x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k}$$

where $\boldsymbol{i}$, $\boldsymbol{j}$, $\boldsymbol{k}$ are “complex” numbers such that
$\boldsymbol{i}^2 = \boldsymbol{j}^2 = \boldsymbol{k}^2 = \boldsymbol{ijk} = -1$

A **quaternion** can therefore be represented as a 4-dimensional vector

$$\mathbf{q} = (w, x, y, z)$$

Quaternions

$x + yi$

The $x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k}$ part is similar to a 3D vector, so we may express a quaternion as

$\mathbb{R}$ $\quad \mathbb{R}^3$

$$\mathbf{q} = (w,\ \mathbf{v})$$

scalar 3D vector

A vector is represented as a quaternion by setting the scalar part 0:

$$\mathbf{q}_{\mathbf{u}} = (0,\ \mathbf{u})$$

Quaternion Normalization

Magnitude:

$$\|\mathbf{q}\| = \sqrt{(w^2 + x^2 + y^2 + z^2)}$$

Normalization:

$$\hat{\mathbf{q}} = \frac{\mathbf{q}}{\|\mathbf{q}\|}$$

Unit Quaternions as Rotations

A unit quaternion is a quaternion $\mathbf{q} = (w, \mathbf{v})$ such that

(w, x, y, z)

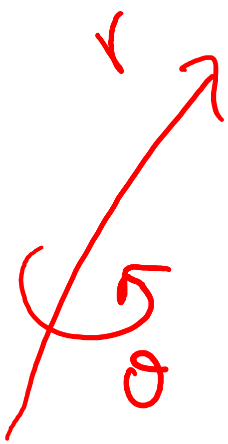
$$w^2 + \mathbf{v} \cdot \mathbf{v} = 1$$

$w^2 + x^2 + y^2 + z^2 = 1$

$\mathbf{q}$ can also be written as

$$\mathbf{q} = (\underbrace{\cos\frac{\theta}{2}}_{w}, \underbrace{\sin\frac{\theta}{2}\mathbf{r}}_{v})$$

$v \cdot v = \sin^2\frac{\theta}{2}\, r \cdot r$



$\mathbf{r}$ is a unit vector representing the axis of rotation

θ is the angle of rotation

Example

$r = (\overset{r_x}{0}, \overset{r_y}{0}, \overset{r_z}{1})$

$\theta = \frac{\pi}{2}$

What is the quaternion representing a rotation about the z-axis by 90 degrees?

$$w = \cos\left(\frac{\pi}{4}\right) = \frac{\sqrt{2}}{2}$$

$$x = \overset{r_x}{0} \cdot \sin\left(\frac{\pi}{4}\right) = 0$$

$$y = \overset{r_y}{0} \cdot \sin\left(\frac{\pi}{4}\right) = 0$$

$$z = \overset{r_z}{1} \cdot \sin\left(\frac{\pi}{4}\right) = \frac{\sqrt{2}}{2}$$

$$\mathbf{q} = \left(\frac{\sqrt{2}}{2}, 0, 0, \frac{\sqrt{2}}{2}\right)$$

Quaternion Operations

For addition and scalar multiplication, a quaternion behaves like a 4-vector:

$$(w_1, x_1, y_1, z_1) + (w_2, x_2, y_2, z_2) = (w_1 + w_2, x_1 + x_2, y_1 + y_2, z_1 + z_2)$$

$$a(w, x, y, z) = (aw, ax, ay, az)$$

Given a quaternion $\mathbf{q}$, what is $-\mathbf{q}$?

(w, x, y, z) $(-w, -x, -y, -z)$

Quaternion Negation

$\cos(\pi + \phi) = -\cos\phi$
$\sin(\pi + \phi) = -\sin\phi$

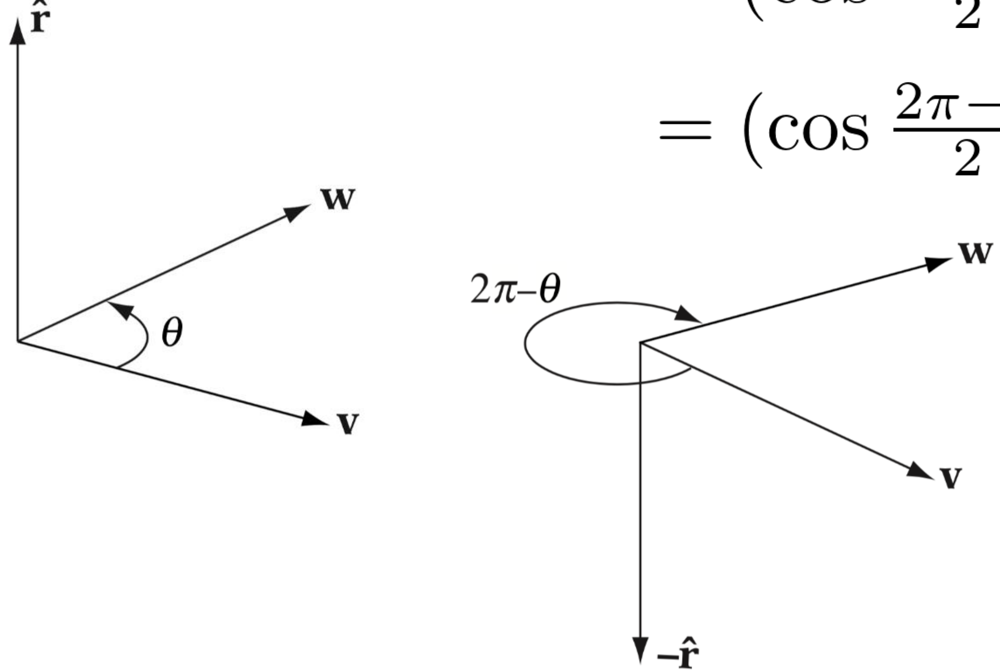
$$\mathbf{q} = \left(\cos\frac{\theta}{2}, \sin\frac{\theta}{2}\mathbf{r}\right)$$

$$-\mathbf{q} = \left(-\cos\frac{\theta}{2}, -\sin\frac{\theta}{2}\mathbf{r}\right)$$

$$= \left(\cos\frac{2\pi+\theta}{2}, \sin\frac{2\pi+\theta}{2}\mathbf{r}\right) \rightarrow ①$$

$$= \left(\cos\frac{2\pi-\theta}{2}, \sin\frac{2\pi-\theta}{2}(-\mathbf{r})\right) \rightarrow ②$$

q



q & -q represent the same orientation

Quaternion Composition

$M = M_{100} \cdots M_3 \cdot M_2 \cdot M_1$

$\cdots q_2 \cdot q_1$

$q = q_{100} \cdots$

Let $\mathbf{q}_1$ and $\mathbf{q}_2$ be two unit quaternions representing two rotations.

$q_1 q_2$

$$\mathbf{q}_1 = (w_1, \mathbf{v}_1) \qquad \mathbf{q}_2 = (w_2, \mathbf{v}_2)$$

$\neq q_2 q_1$

The composition of first a rotation by $\mathbf{q}_1$ and then a rotation by $\mathbf{q}_2$ is given by the multiplication of $\mathbf{q}_2$ and $\mathbf{q}_1$:

$$\mathbf{q}_2\mathbf{q}_1 = (\overset{1}{w_1 w_2} - \overset{3}{\mathbf{v}_1 \cdot \mathbf{v}_2},\ \overset{3}{w_1\mathbf{v}_2} + \overset{3}{w_2\mathbf{v}_1} + \overset{6}{\mathbf{v}_2 \times \mathbf{v}_1})$$

Order matters!

Vector dot product

Vector cross product

Compositing two rotations using quaternions take 16 multiplications and 12 additio

Quaternion Inverse

negation : $-q$

The inverse of a quaternion $\mathbf{q}$ is denoted by $\mathbf{q}^{-1}$, such that

$$\mathbf{q}\mathbf{q}^{-1} = (1, 0, 0, 0)$$

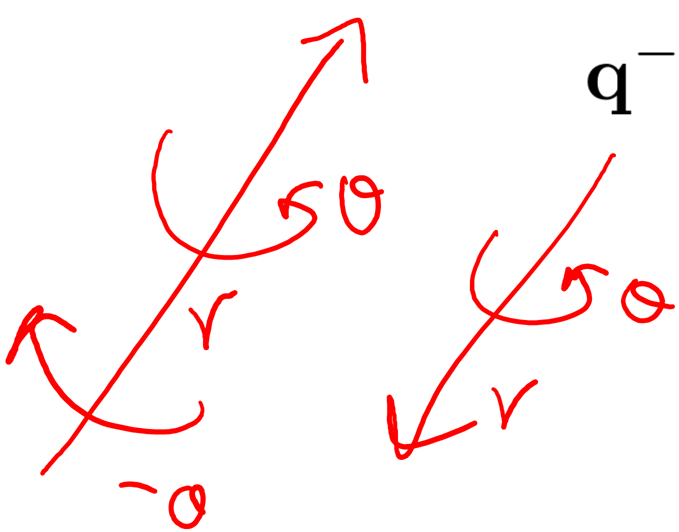
Identity quaternion, also representing zero rotation

$q_2 \cdot q_1 = (. \; . \; . \; . \; — \; .)$

Given $\mathbf{q} = (w, \mathbf{v})$, what is $\mathbf{q}^{-1}$?

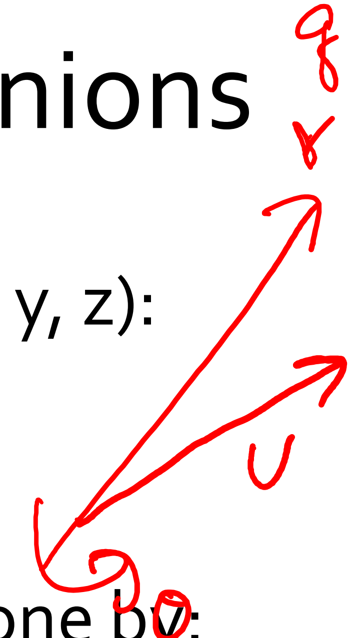
$$\mathbf{q}^{-1} = (w, -\mathbf{v})$$

Negating the axis of rotation



Inverting a quaternion is fast!

Rotating Vectors with Quaternions



Let v be a quaternion representing a vector (x, y, z):

$$\mathbf{v} = (0, x, y, z)$$

Rotating a vector **v** by a unit quaternion **q** is done by:

$$\mathbf{v}' = \mathbf{q}\mathbf{v}\mathbf{q}^{-1}$$

$p \qquad p^{-1}$

Further apply a rotation by a unit quaternion **p**:

$$\mathbf{v}'' = \mathbf{p}\mathbf{q}\mathbf{v}\mathbf{q}^{-1}\mathbf{p}^{-1} = \mathbf{p}\mathbf{q}\mathbf{v}(\mathbf{p}\mathbf{q})^{-1}$$

$\mathbf{pq}$ is the composite rotation