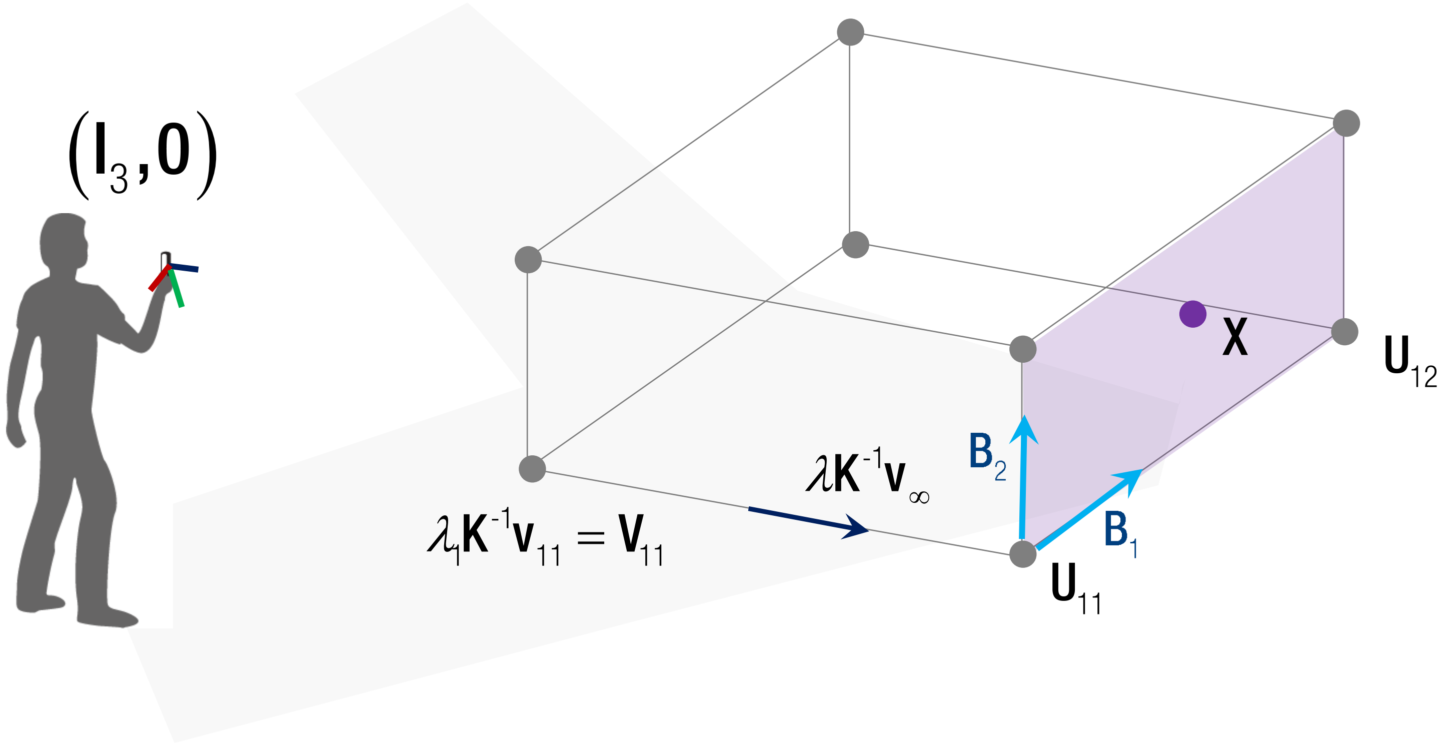
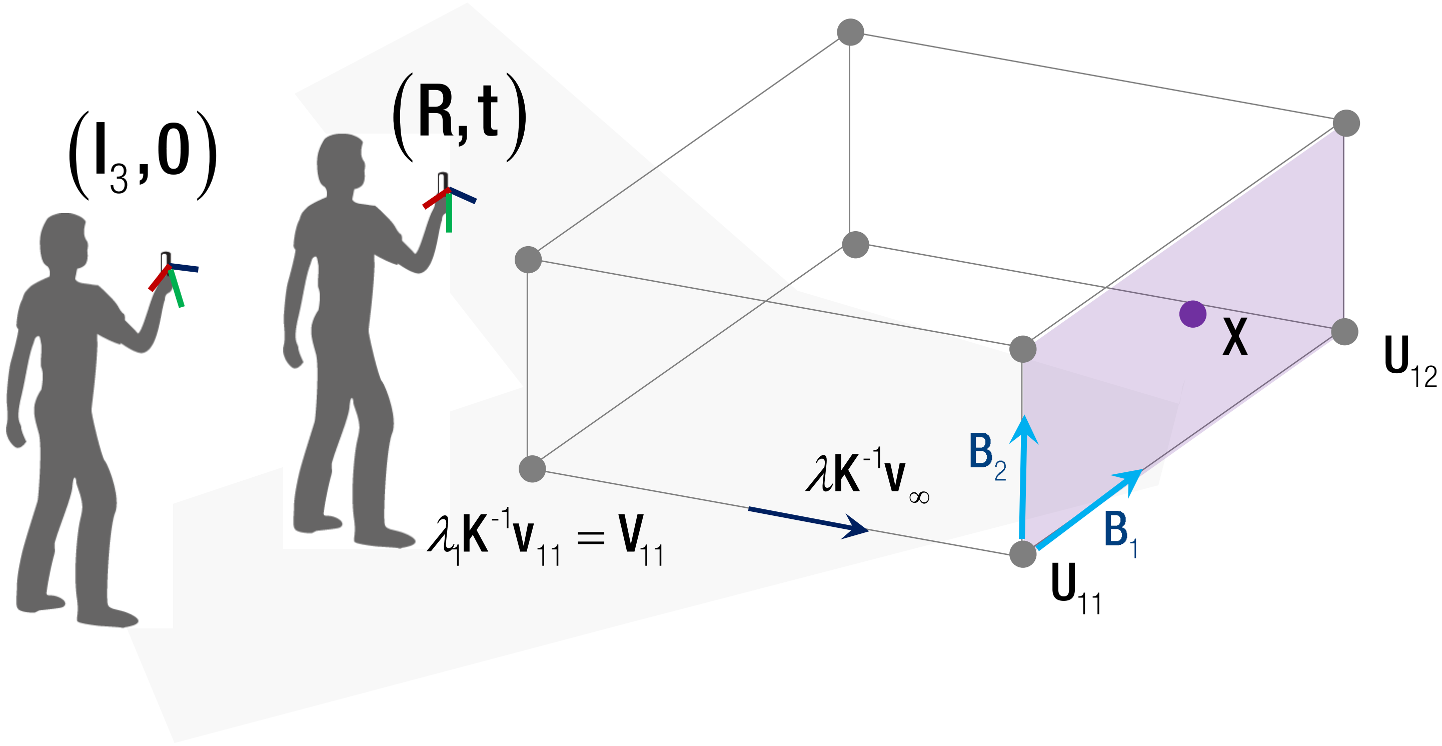
Rotation

Hyun Soo Park

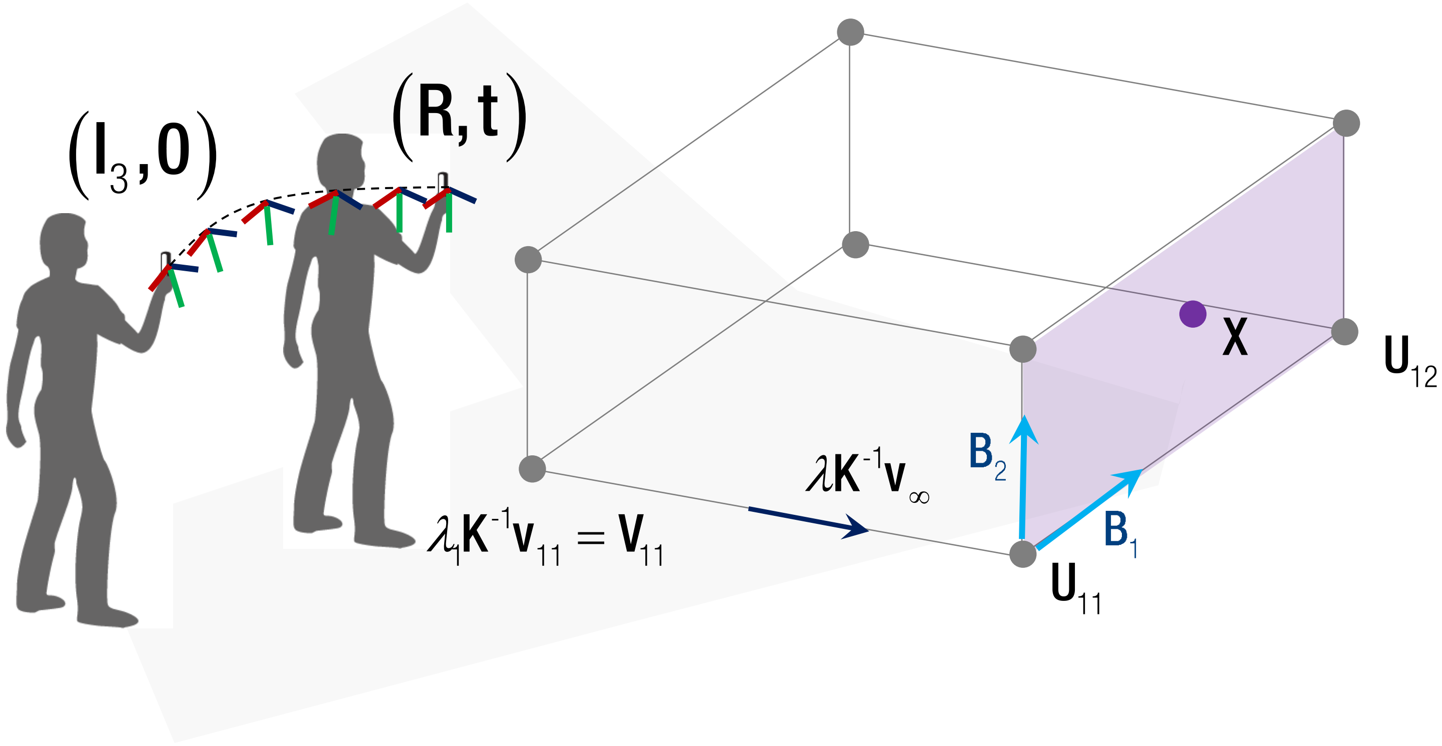
Interpolation of Transformation



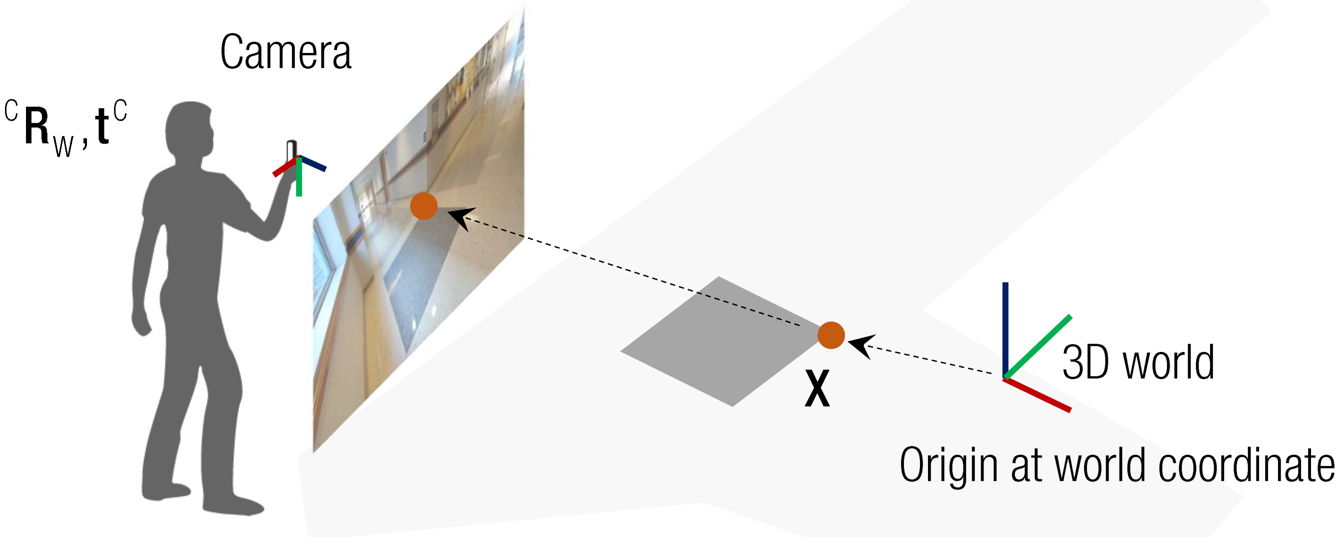
Interpolation of Transformation

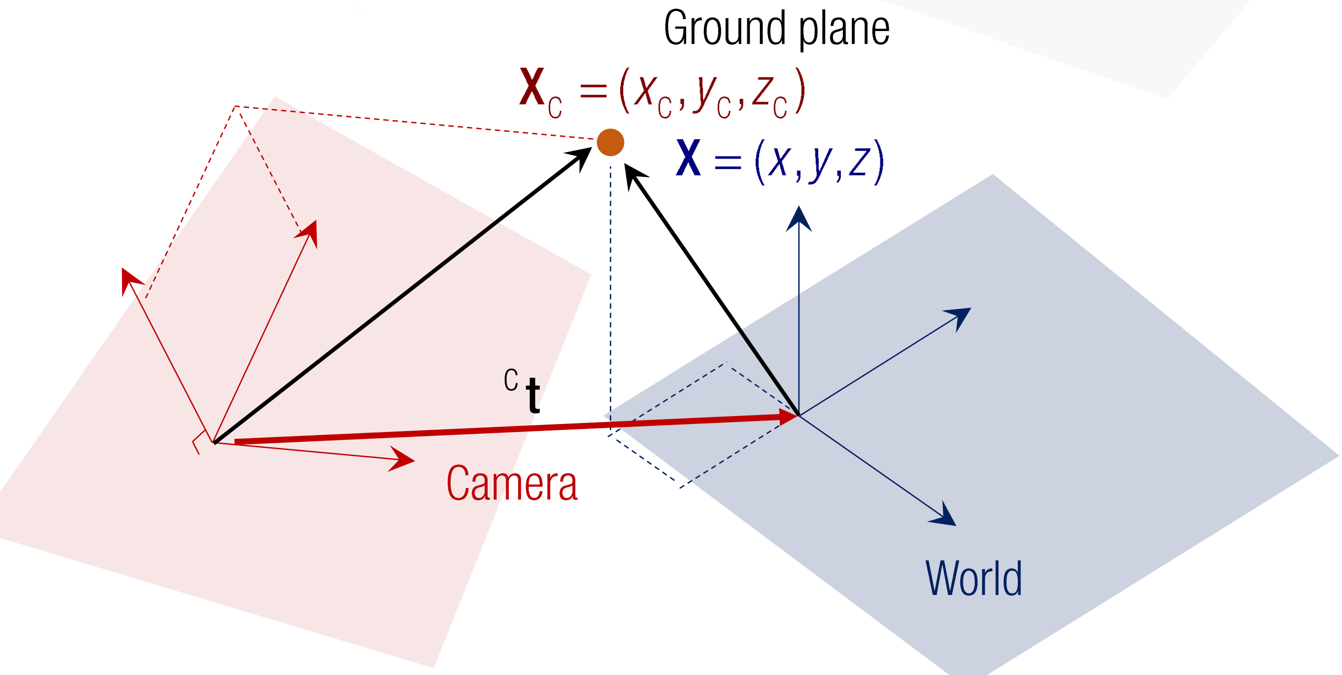


Interpolation of Transformation



Recall: Rotate and then, Translate



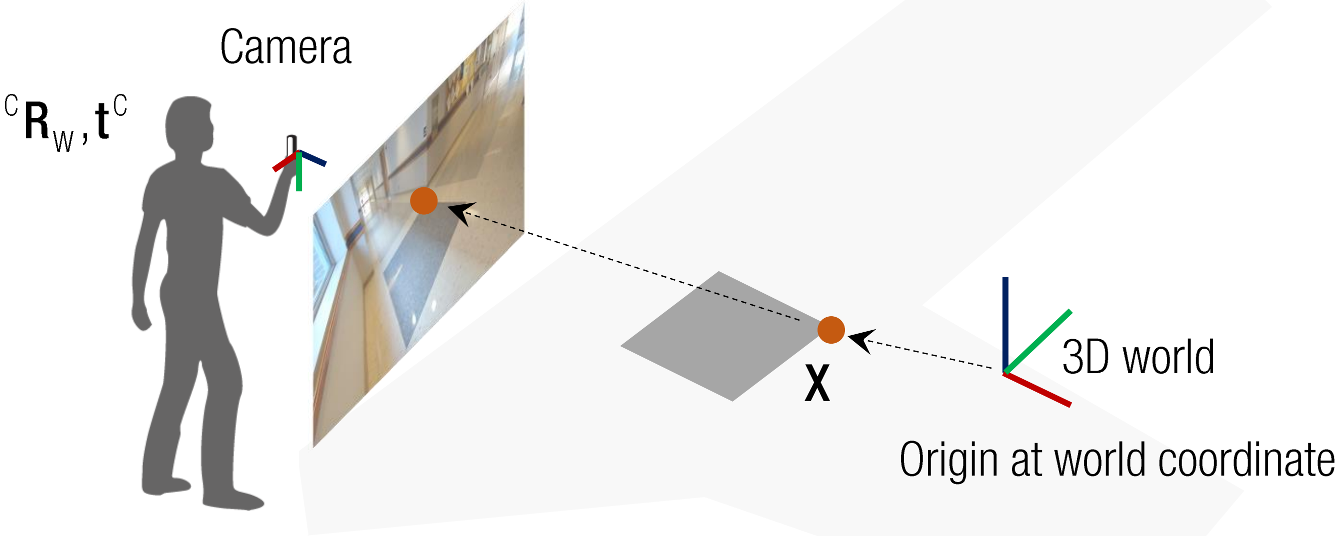


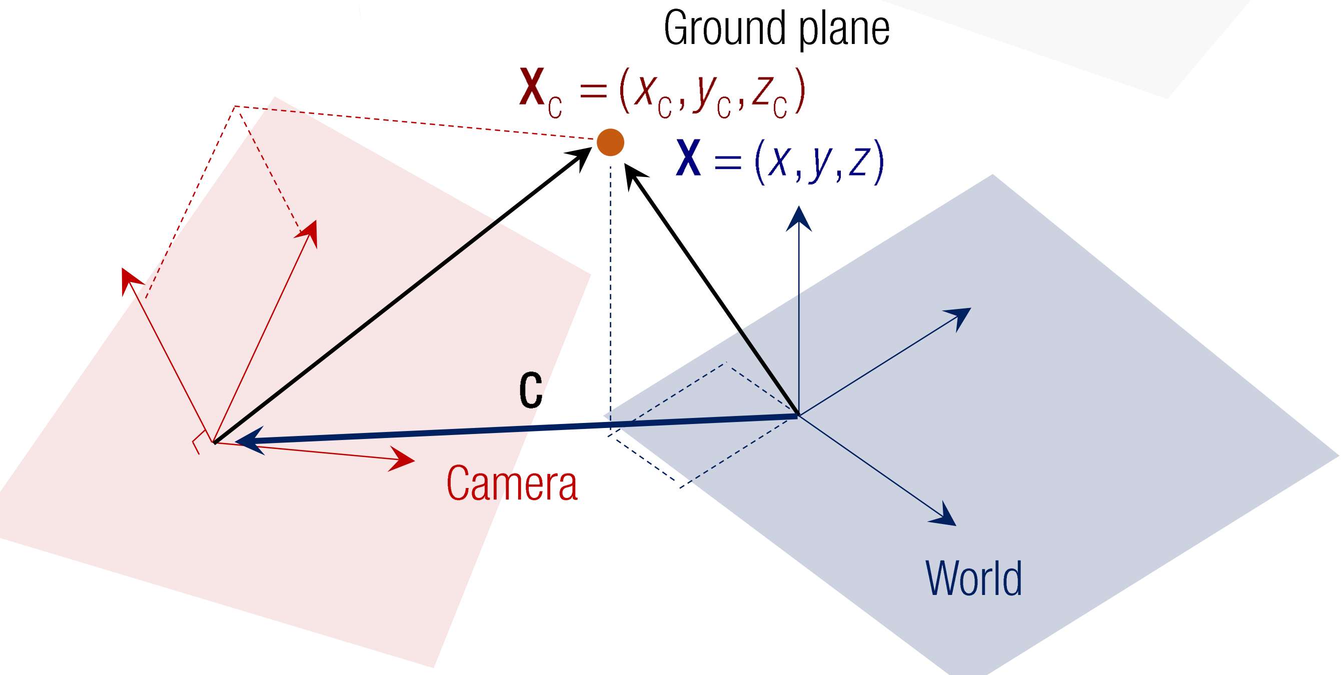
$$\mathbf{X}_C = {}^{C}\mathbf{R}_W \mathbf{X} + {}^{C}\mathbf{t} = \begin{bmatrix} r_{x1} & r_{x2} & r_{x3} & t_x \\ r_{y1} & r_{y2} & r_{y3} & t_y \\ r_{z1} & r_{z2} & r_{z3} & t_z \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

where $^{C}\mathbf{t}$ is translation from world to camera seen from camera.

Rotate and then, translate.

Recall: Translate and the, Rotate





$$\mathbf{X}_C = {}^{C}\mathbf{R}_W \mathbf{X} + {}^{C}\mathbf{t} = \begin{bmatrix} r_{x1} & r_{x2} & r_{x3} & t_x \\ r_{y1} & r_{y2} & r_{y3} & t_y \\ r_{z1} & r_{z2} & r_{z3} & t_z \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

where ${}^{C}\mathbf{t}$ is translation from world to camera seen from camera.

Rotate and then, translate.

---

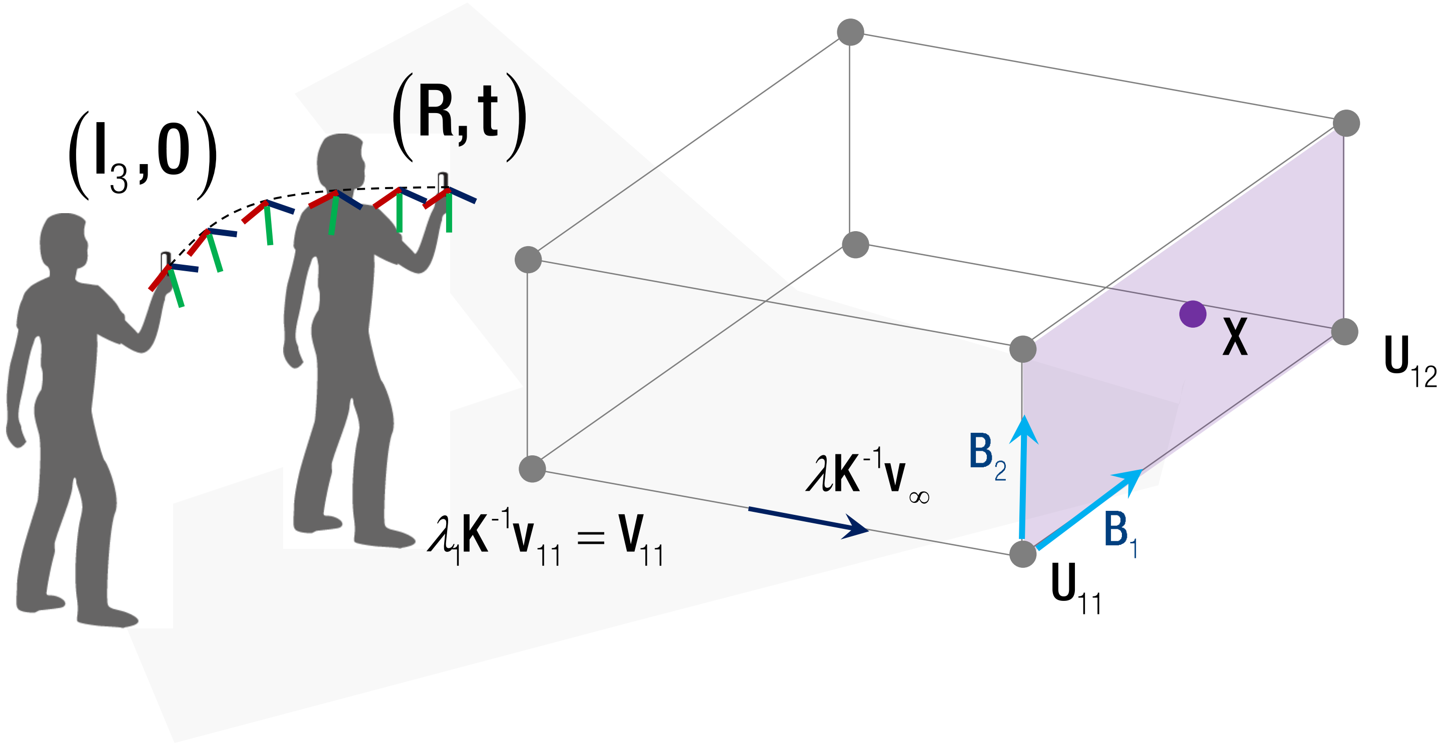
cf) **Translate and then, rotate.**

$$\mathbf{X}_C = {}^{C}\mathbf{R}_W (\mathbf{X} - \mathbf{C}) = \begin{bmatrix} r_{x1} & r_{x2} & r_{x3} \\ r_{y1} & r_{y2} & r_{y3} \\ r_{z1} & r_{z2} & r_{z3} \end{bmatrix} \begin{bmatrix} 1 & & & -C_x \\ & 1 & & -C_y \\ & & 1 & -C_z \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$

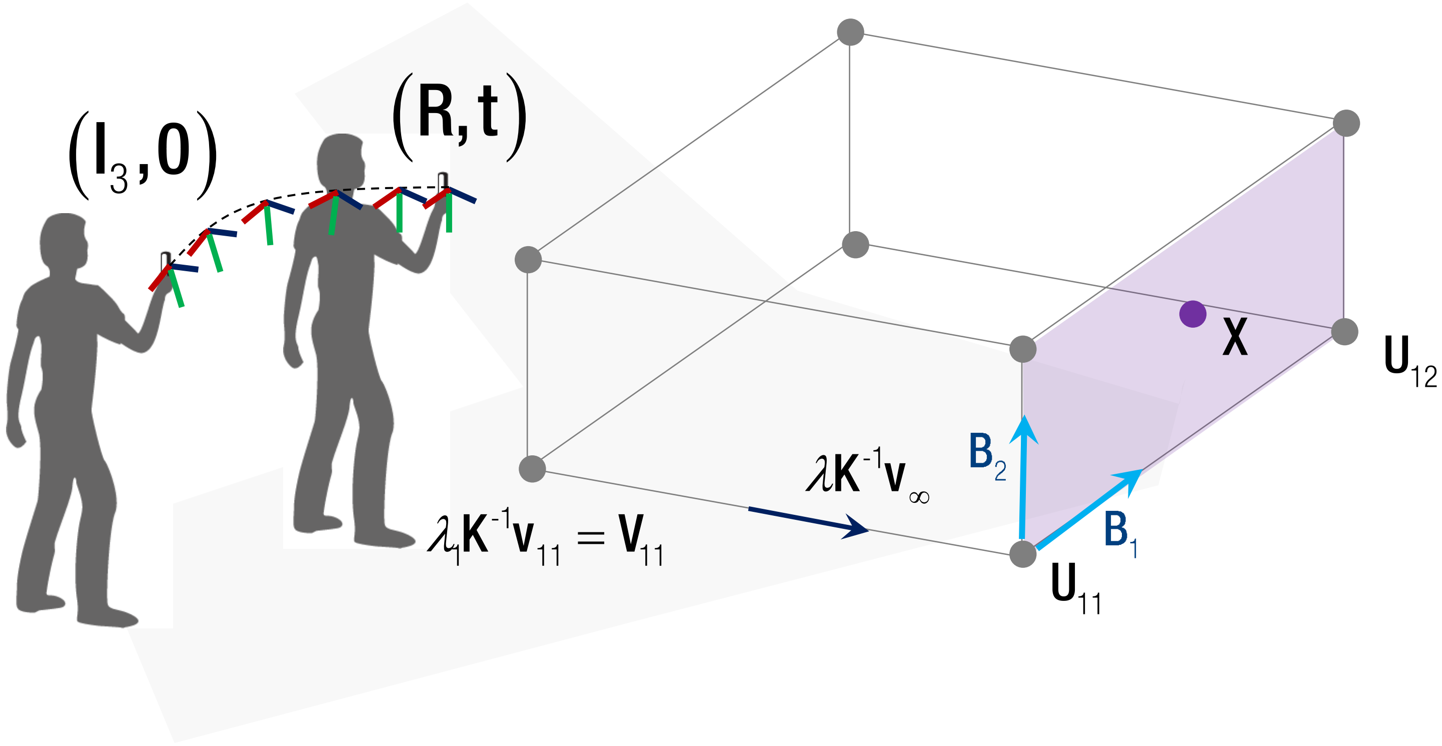
where $\mathbf{C}$ is translation from world to camera seen from world.

Interpolation of Translation

$$\lambda\tilde{\mathbf{u}} = \mathbf{KR}\begin{bmatrix}\mathbf{I}_3 & -\mathbf{C}\end{bmatrix}\begin{bmatrix}X\\Y\\Z\\1\end{bmatrix} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\begin{bmatrix}X\\Y\\Z\\1\end{bmatrix}$$



Interpolation of Translation



$$\lambda \tilde{\mathbf{u}} = \mathbf{KR}\begin{bmatrix}\mathbf{I}_3 & -\mathbf{C}\end{bmatrix}\begin{bmatrix}X\\Y\\Z\\1\end{bmatrix} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\begin{bmatrix}X\\Y\\Z\\1\end{bmatrix}$$

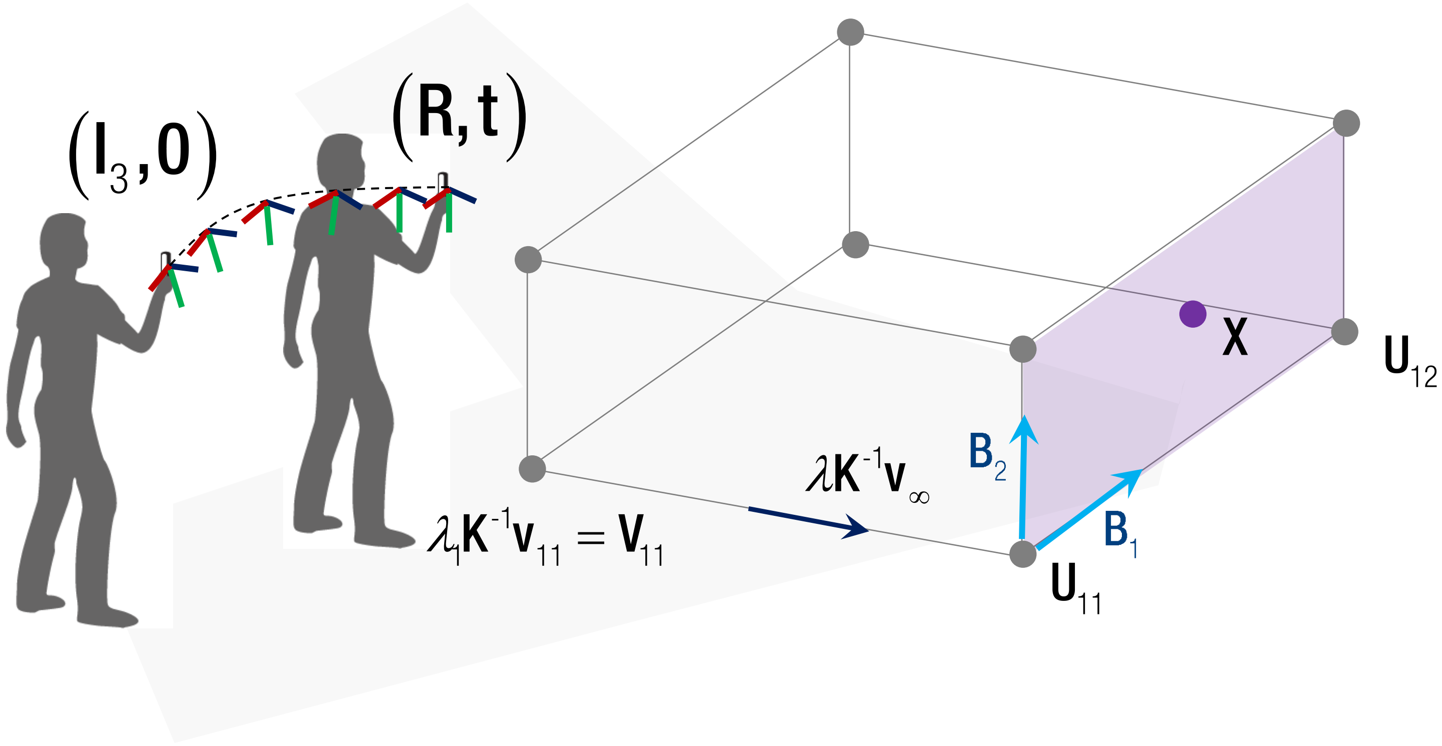
Rot. ➔ Trans. Trans. ➔ Rot.

Translation is independent on rotation.

How to interpolate translation?

$$\mathbf{C}_1 = \begin{bmatrix}C_1^x\\C_1^y\\C_1^z\end{bmatrix} \rightarrow \mathbf{C}_2 = \begin{bmatrix}C_2^x\\C_2^y\\C_2^z\end{bmatrix}$$

Interpolation of Translation



$$\lambda \tilde{\mathbf{u}} = \mathbf{KR}\begin{bmatrix} \mathbf{I}_3 & -\mathbf{C} \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{K}\begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Rot. ➔ Trans. Trans. ➔ Rot.

Translation is independent on rotation.

How to interpolate translation?

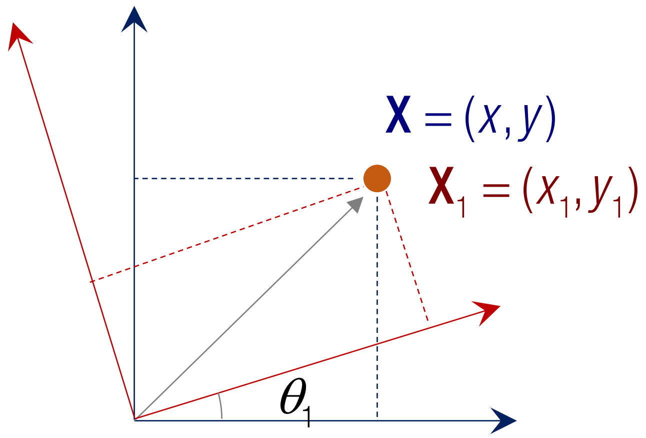
$$\mathbf{C}_1 = \begin{bmatrix} C_1^x \\ C_1^y \\ C_1^z \end{bmatrix} \rightarrow \mathbf{C}_2 = \begin{bmatrix} C_2^x \\ C_2^y \\ C_2^z \end{bmatrix}$$

Interpolated camera center:

$$\mathbf{C}_i = w\mathbf{C}_1 + (1-w)\mathbf{C}_2 \qquad w \in [0,1]$$

Interpolation of Rotation

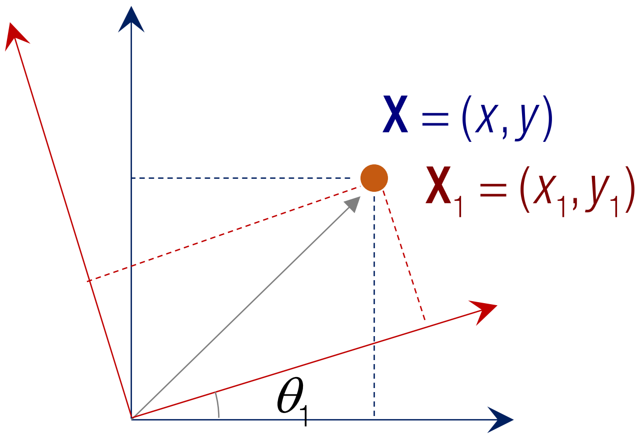
2D coordinate transform:



$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \qquad \begin{bmatrix} x \\ y \end{bmatrix}$$

Interpolation of Rotation

2D coordinate transform:

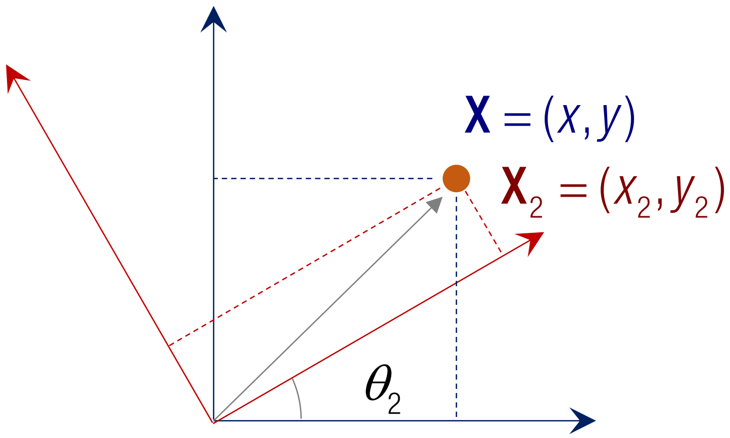


$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\det\left(\begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{bmatrix}\right) = \cos^2\theta_1 + \sin^2\theta_1 = 1$$

Interpolation of Rotation

2D coordinate transform:



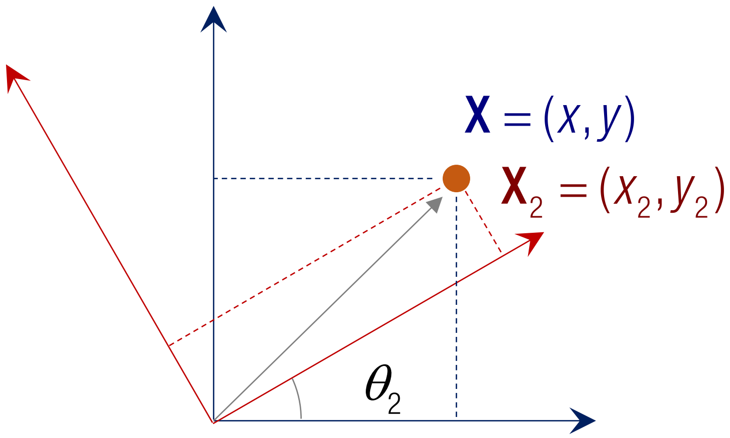
$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\downarrow$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} \cos\theta_2 & \sin\theta_2 \\ -\sin\theta_2 & \cos\theta_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Interpolation of Rotation

2D coordinate transform:



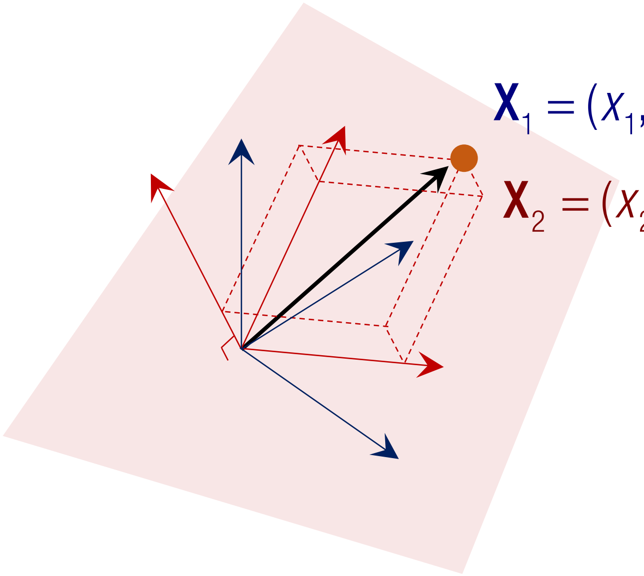
$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\theta = w\theta_1 + (1-w)\theta_2$$

$$w \in [0,1]$$

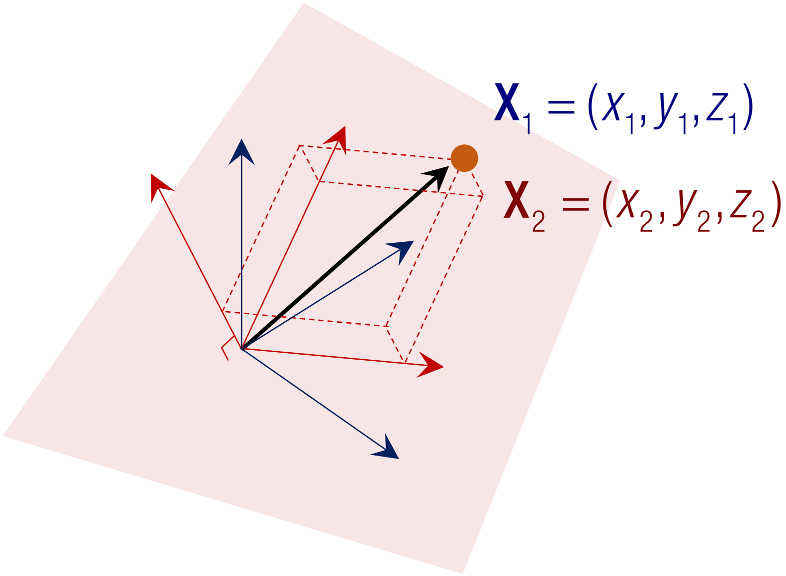
$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} \cos\theta_2 & \sin\theta_2 \\ -\sin\theta_2 & \cos\theta_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Interpolation of Rotation in 3D



$$\mathbf{X}_2 = \begin{bmatrix} r_{x1} & r_{x2} & r_{x3} \\ r_{y1} & r_{y2} & r_{y3} \\ r_{z1} & r_{z2} & r_{z3} \end{bmatrix} \mathbf{X} = \mathbf{R}_1 \mathbf{X}_1$$

Interpolation of Rotation in 3D



$$\mathbf{X}_2 = \begin{bmatrix} r_{x1} & r_{x2} & r_{x3} \\ r_{y1} & r_{y2} & r_{y3} \\ r_{z1} & r_{z2} & r_{z3} \end{bmatrix} \mathbf{X} = \mathbf{R}_1 \mathbf{X}_1$$

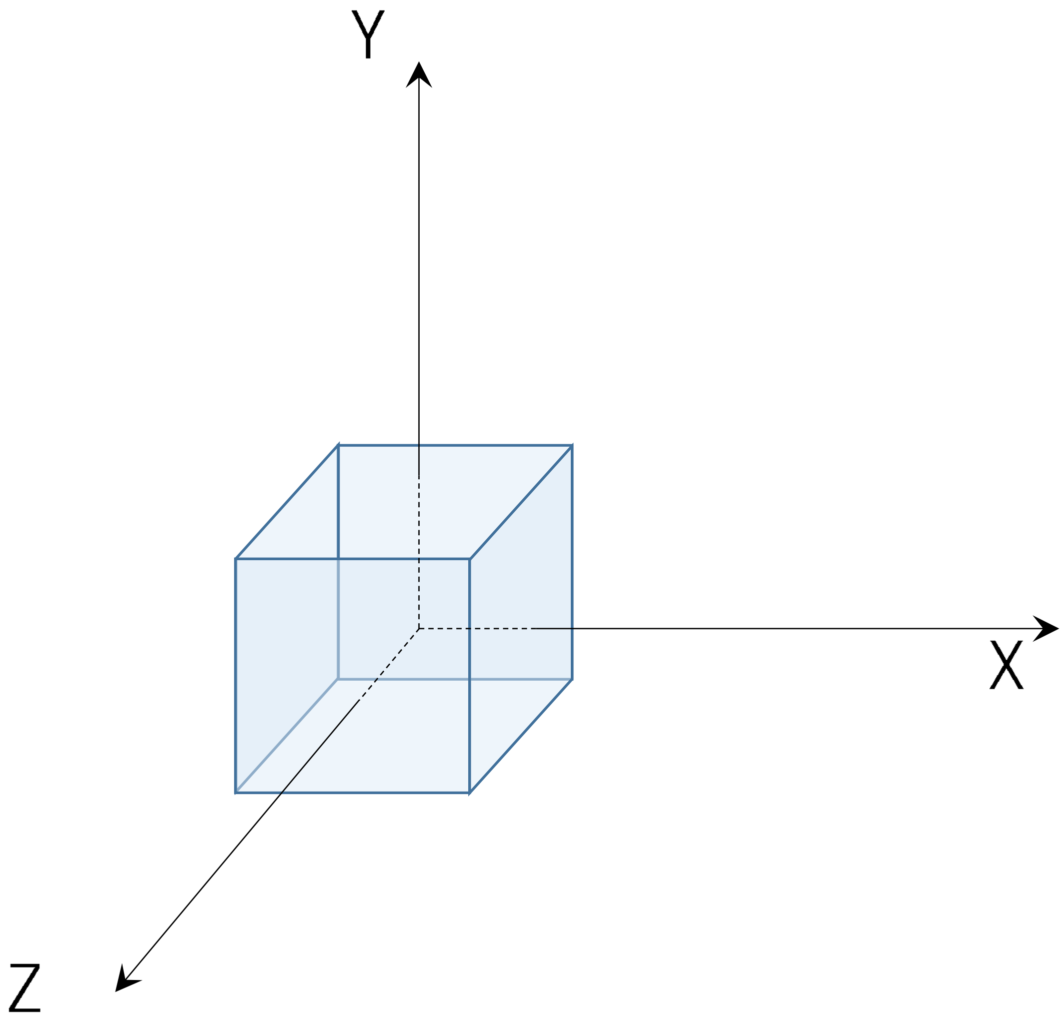
How to interpolate between two coordinates?

$$\mathbf{R}_1 \rightarrow \mathbf{R}_2$$

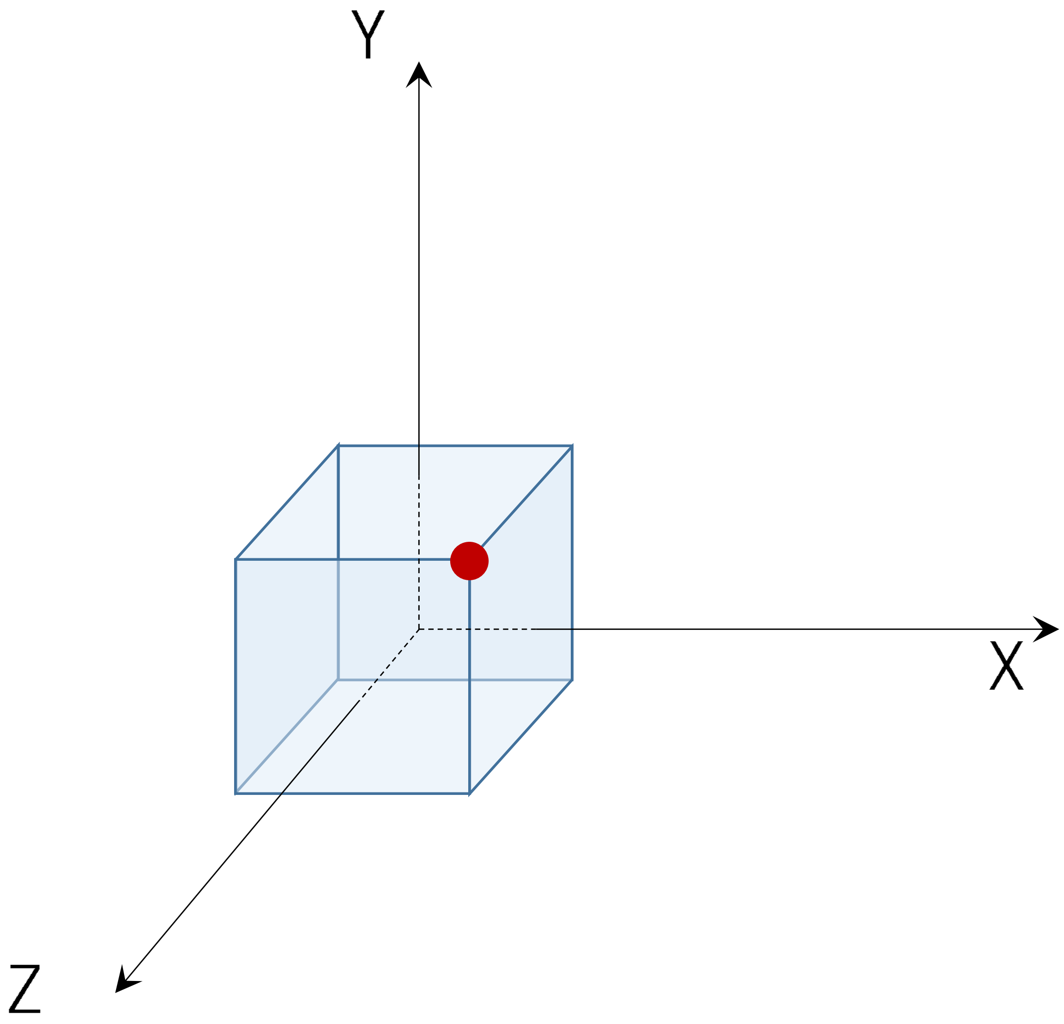
dof: 3

of parameters: 9

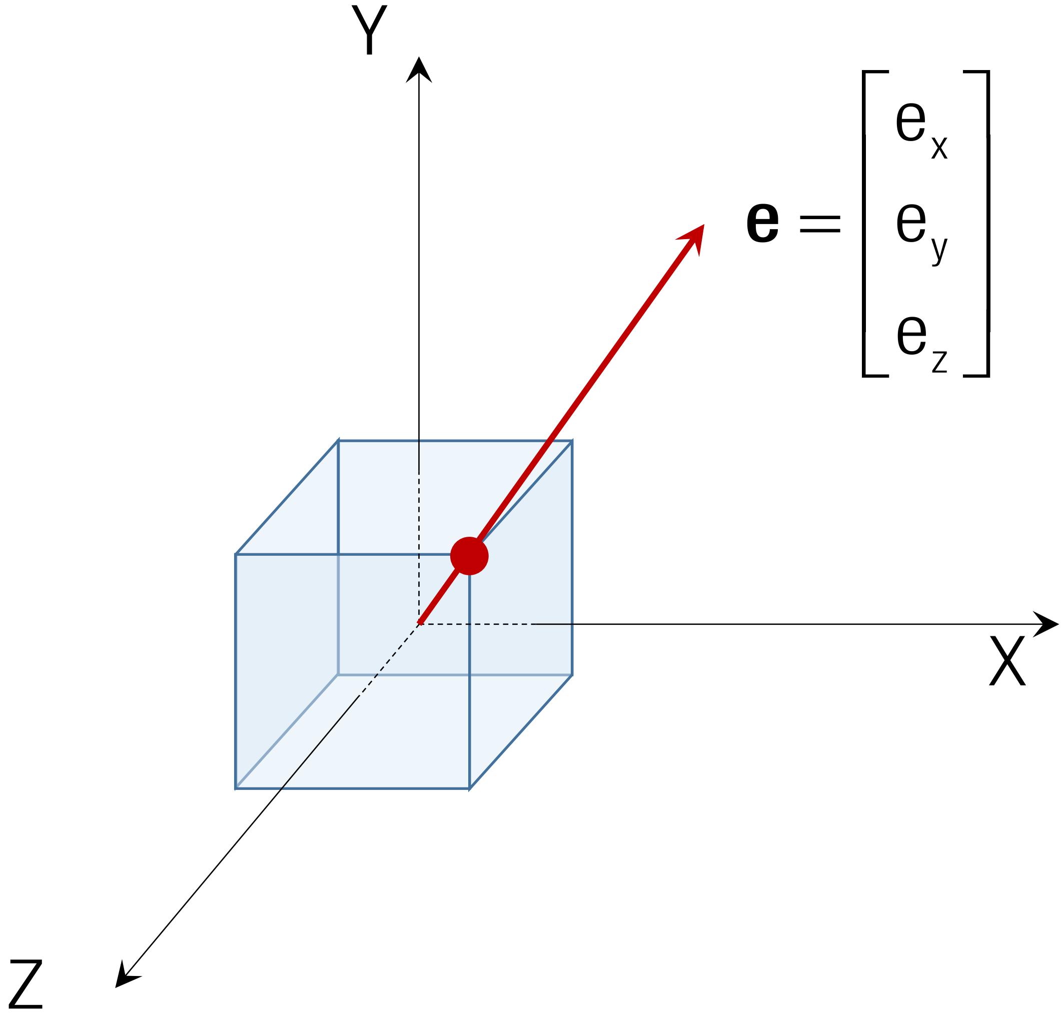
Axis Angle Representation



Axis Angle Representation

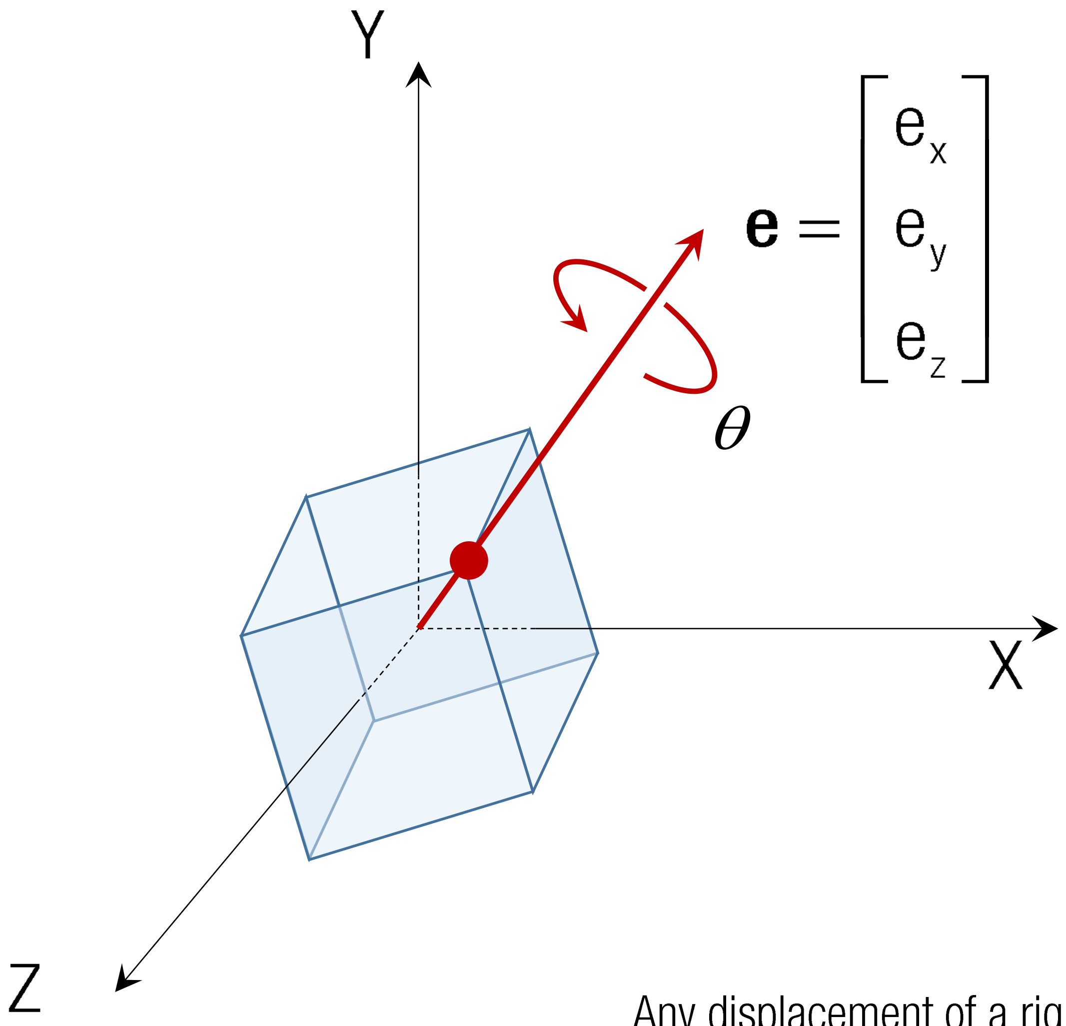


Axis Angle Representation



$$\mathbf{e} = \begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix}$$

Axis Angle Representation

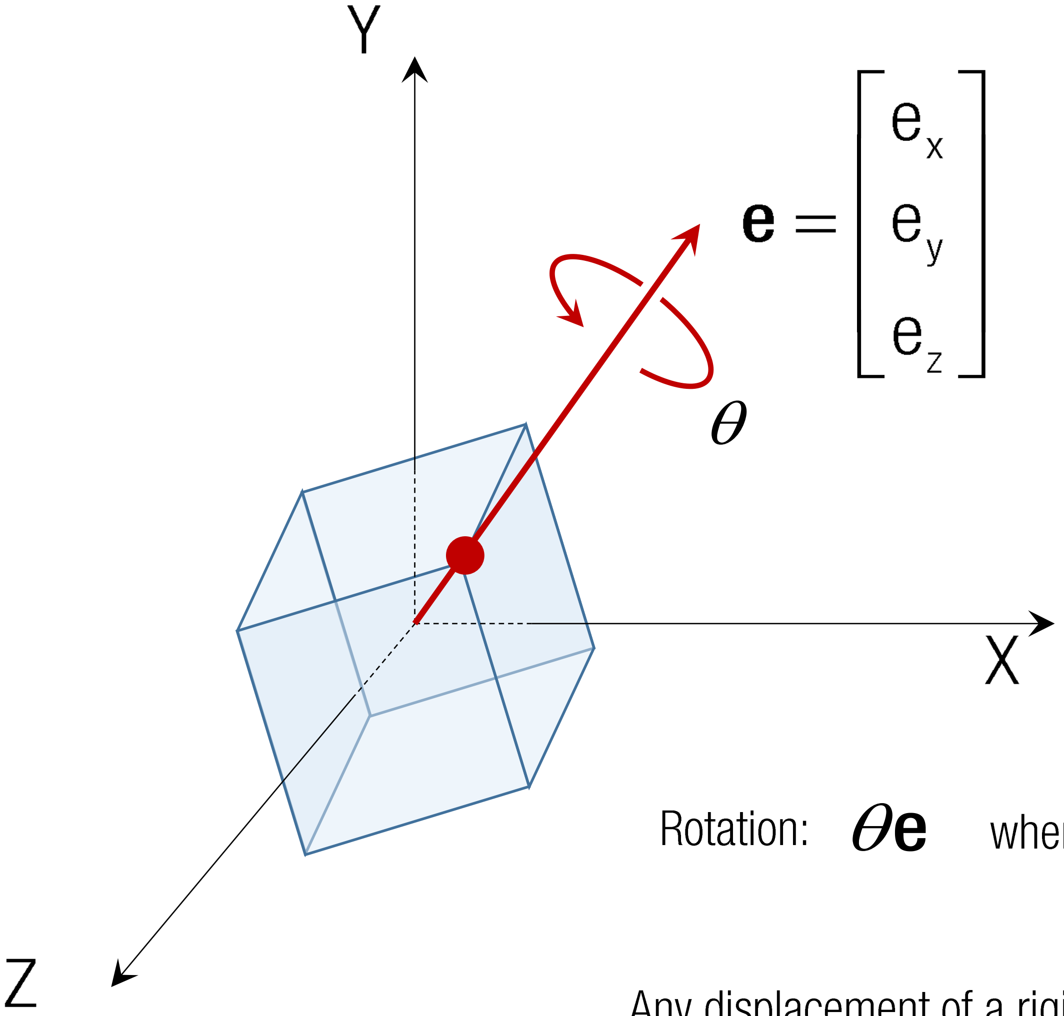


$$\mathbf{e} = \begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix}$$

Euler's theorem

Any displacement of a rigid body such that a point on the rigid body remains fixed, is equivalent to a single rotation about some axis that runs through the fixed point.

Axis Angle Representation



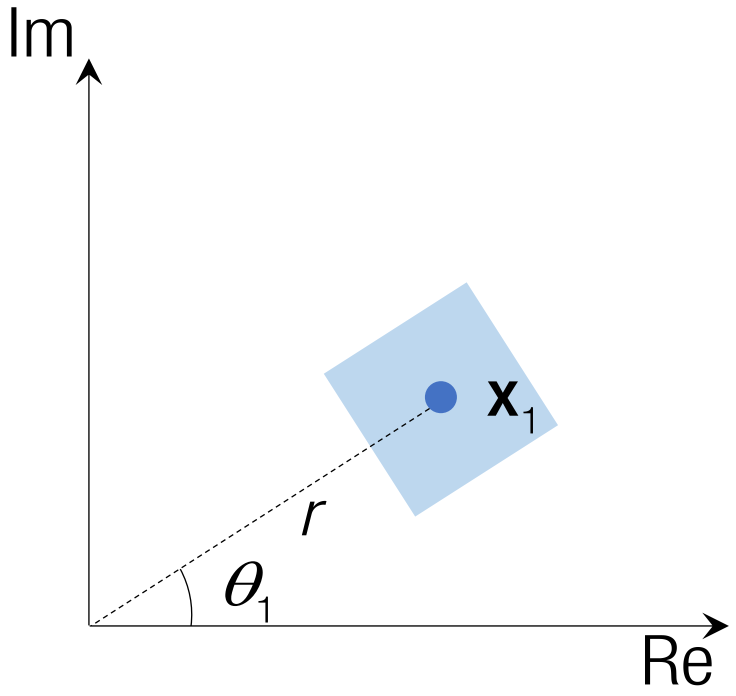
$$\mathbf{e} = \begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix}$$

Rotation: $\theta\mathbf{e}$ where $\|\mathbf{e}\| = 1$

Euler's theorem

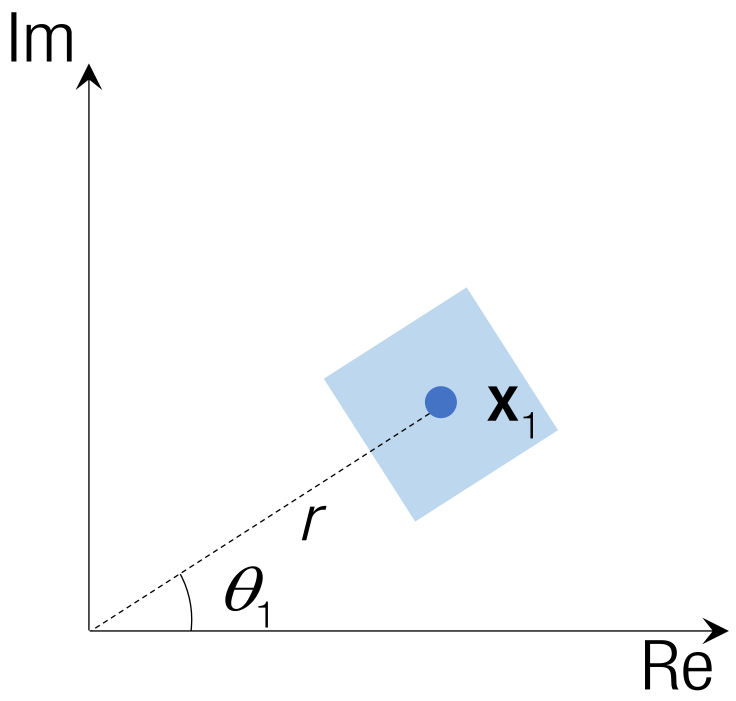
Any displacement of a rigid body such that a point on the rigid body remains fixed, is equivalent to a single rotation about some axis that runs through the fixed point.

2D Exponential Map (Euler's Formula)



$$\mathbf{x}_1 =$$

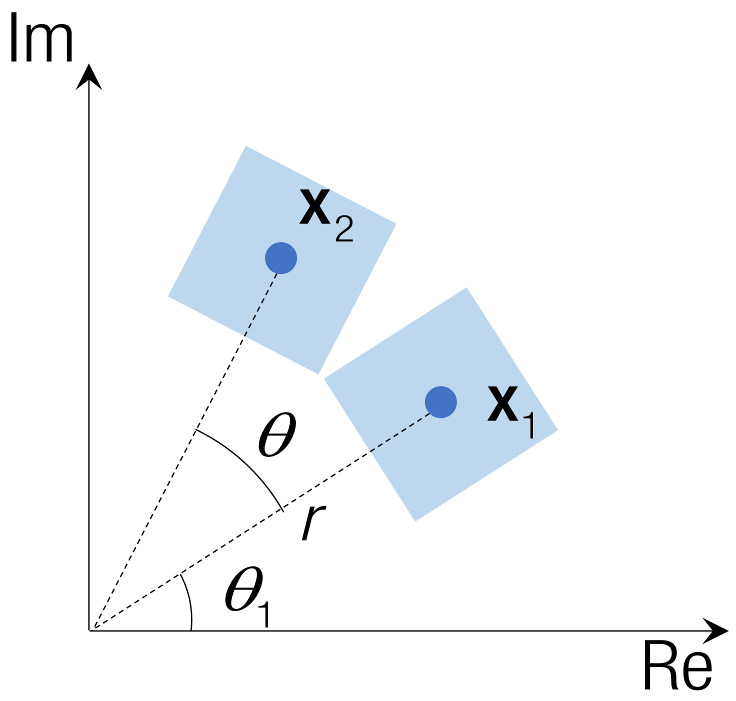
2D Exponential Map (Euler's Formula)



$$\mathbf{x}_1 = r\exp(i\theta_1) = r\left(\cos\theta_1 + i\sin\theta_1\right)$$

Ref) $\exp(x) = \sum_{k=0}^{\infty} \frac{x^k}{k!}$

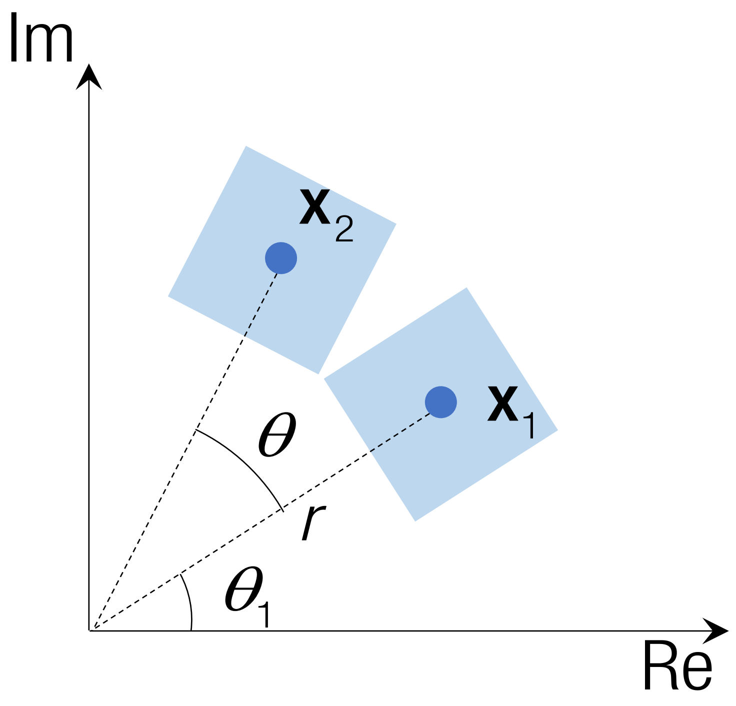
2D Exponential Map (Euler's Formula)



$$\mathbf{x}_1 = r\exp(i\theta_1) = r\left(\cos\theta_1 + i\sin\theta_1\right)$$

$\mathbf{x}_2$

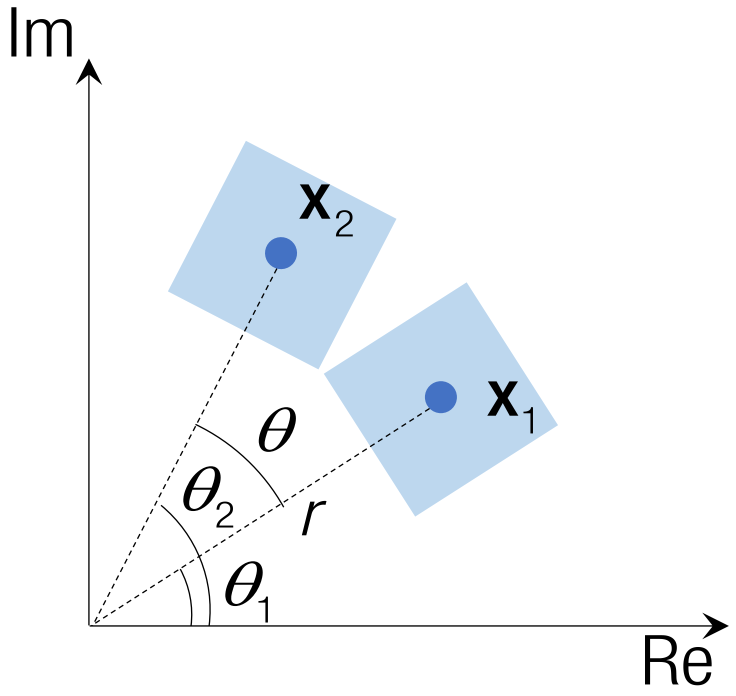
2D Exponential Map (Euler's Formula)



$$\mathbf{x}_1 = r\exp(i\theta_1) = r\left(\cos\theta_1 + i\sin\theta_1\right)$$

$$\begin{aligned}\mathbf{x}_2 &= \exp(i\theta)\mathbf{x}_1 = r\left(\cos\theta + i\sin\theta\right)\left(\cos\theta_1 + i\sin\theta_1\right)\\ &= r\left(\cos\theta\cos\theta_1 - \sin\theta\sin\theta_1 + i\left(\cos\theta\sin\theta_1 + \sin\theta\cos\theta_1\right)\right)\\ &= r\left(\cos(\theta+\theta_1) + i\sin(\theta+\theta_1)\right)\end{aligned}$$

2D Exponential Map (Euler's Formula)

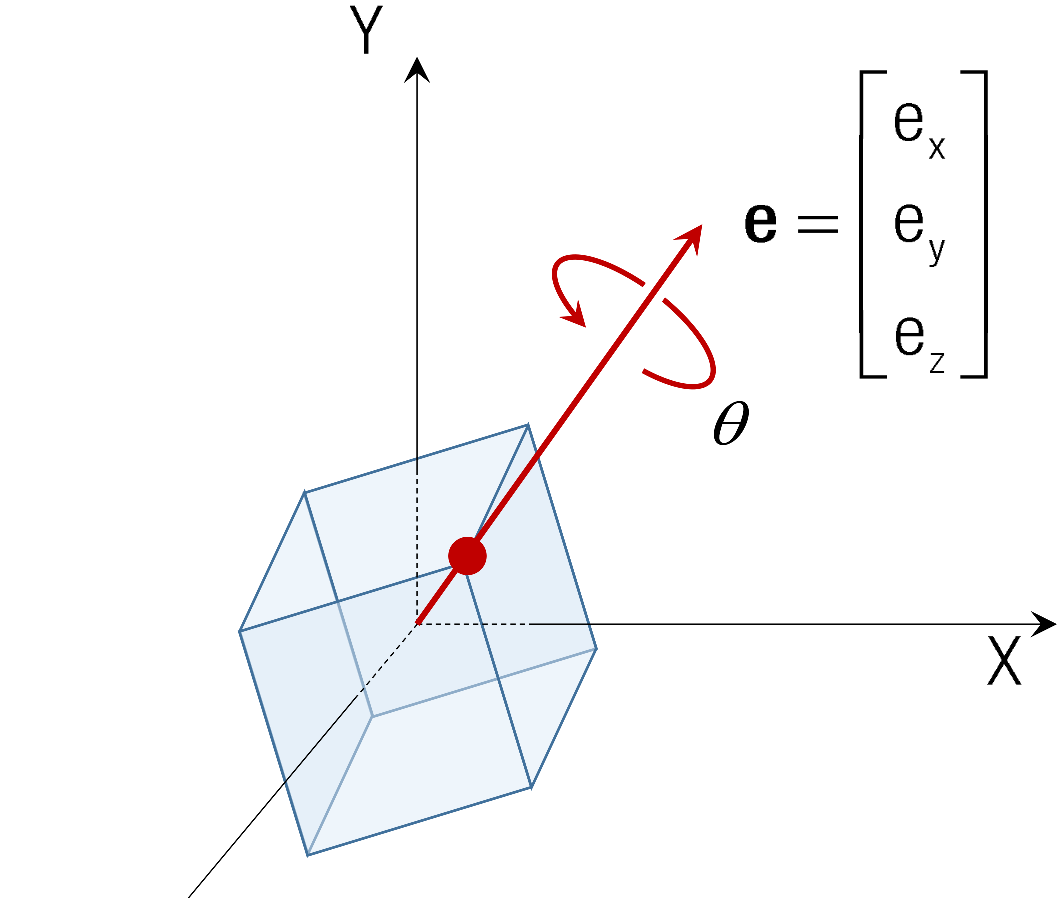


$$\mathbf{x}_1 = r\exp(i\theta_1) = r\left(\cos\theta_1 + i\sin\theta_1\right)$$

$$\begin{aligned}
\mathbf{x}_2 &= \exp(i\theta)\mathbf{x}_1 = r\left(\cos\theta + i\sin\theta\right)\left(\cos\theta_1 + i\sin\theta_1\right) \\
&= r\left(\cos\theta\cos\theta_1 - \sin\theta\sin\theta_1 + i\left(\cos\theta\sin\theta_1 + \sin\theta\cos\theta_1\right)\right) \\
&= r\left(\cos(\theta+\theta_1) + i\sin(\theta+\theta_1)\right) \\
&= r\left(\cos\theta_2 + i\sin\theta_2\right)
\end{aligned}$$

$$\theta_2 = \theta_1 + \theta$$

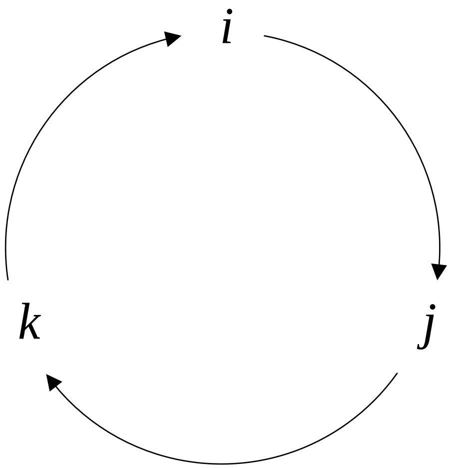
3D Exponential Map: Quaternion



Rotation: $\theta\mathbf{e}$ where $\|\mathbf{e}\| = 1$

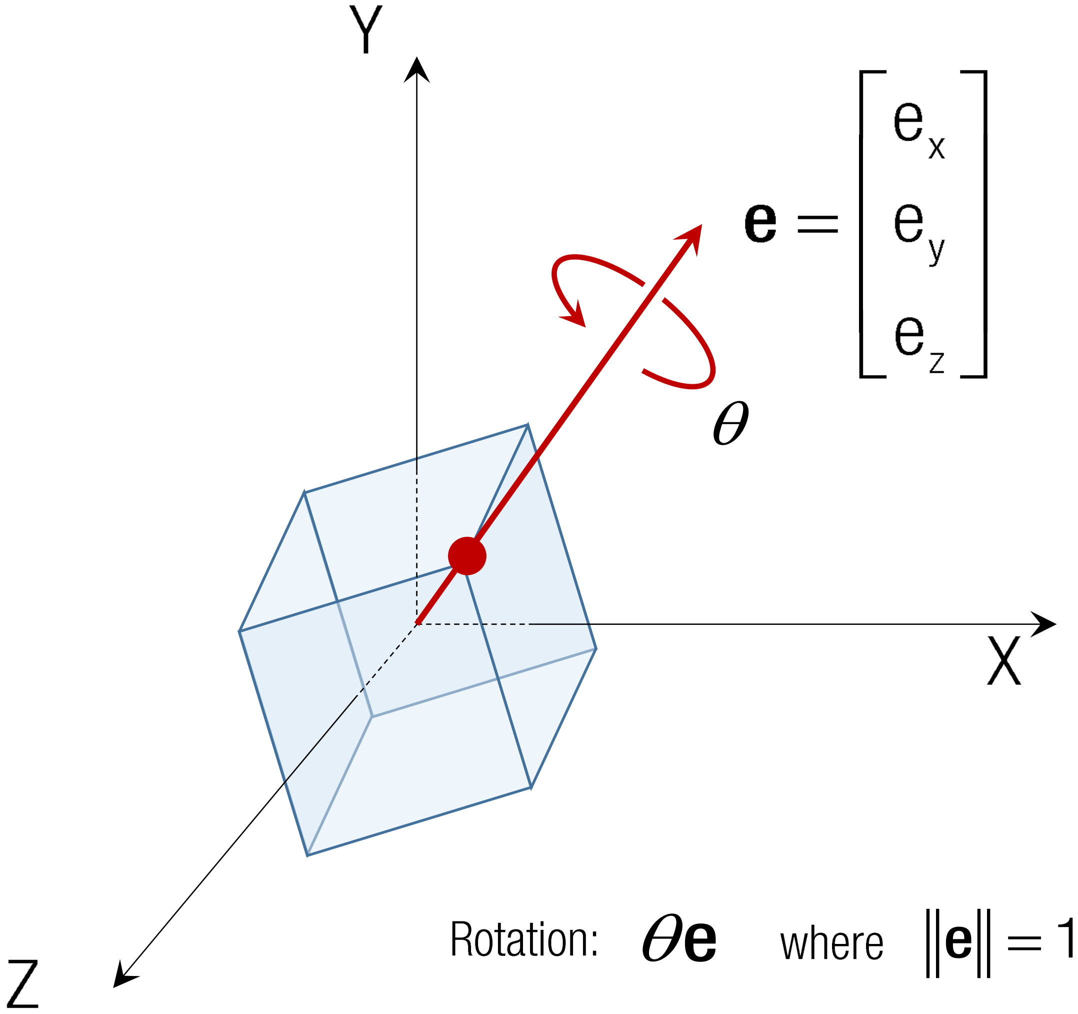
$$\exp\left(\frac{\theta}{2}\mathbf{e}\right) = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\left(i\,e_x + j\,e_y + k\,e_z\right)$$

$$i^2 = j^2 = k^2 = ijk = -1$$



$$ij = k$$
$$jk = i$$
$$ki = j$$

Exercise



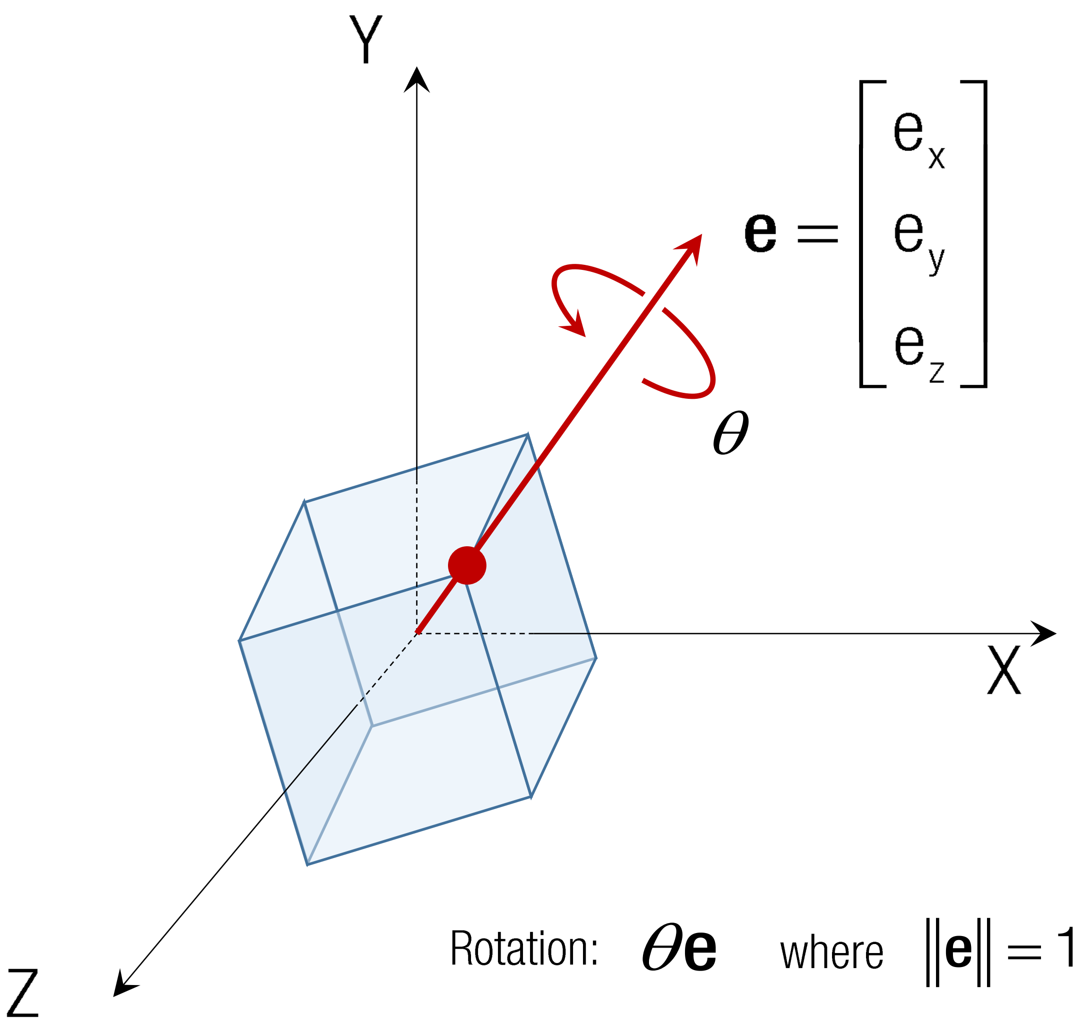
$$\mathbf{e} = \begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix}$$

Rotation: $\theta\mathbf{e}$ where $\|\mathbf{e}\| = 1$

$$\exp\left(\frac{\theta}{2}\mathbf{e}\right) = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\left(i\,e_x + j\,e_y + k\,e_z\right)$$

Find a quaternion **q** such that it describes a rotation of 60 degrees about the axis **a**=[3, 4, 0].

Exercise



Rotation: $\theta\mathbf{e}$ where $\|\mathbf{e}\| = 1$

$$\exp\left(\frac{\theta}{2}\mathbf{e}\right) = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\left(i\,e_x + j\,e_y + k\,e_z\right)$$

Find a quaternion **q** such that it describes a rotation of 60 degrees about the axis **a**=[3, 4, 0].

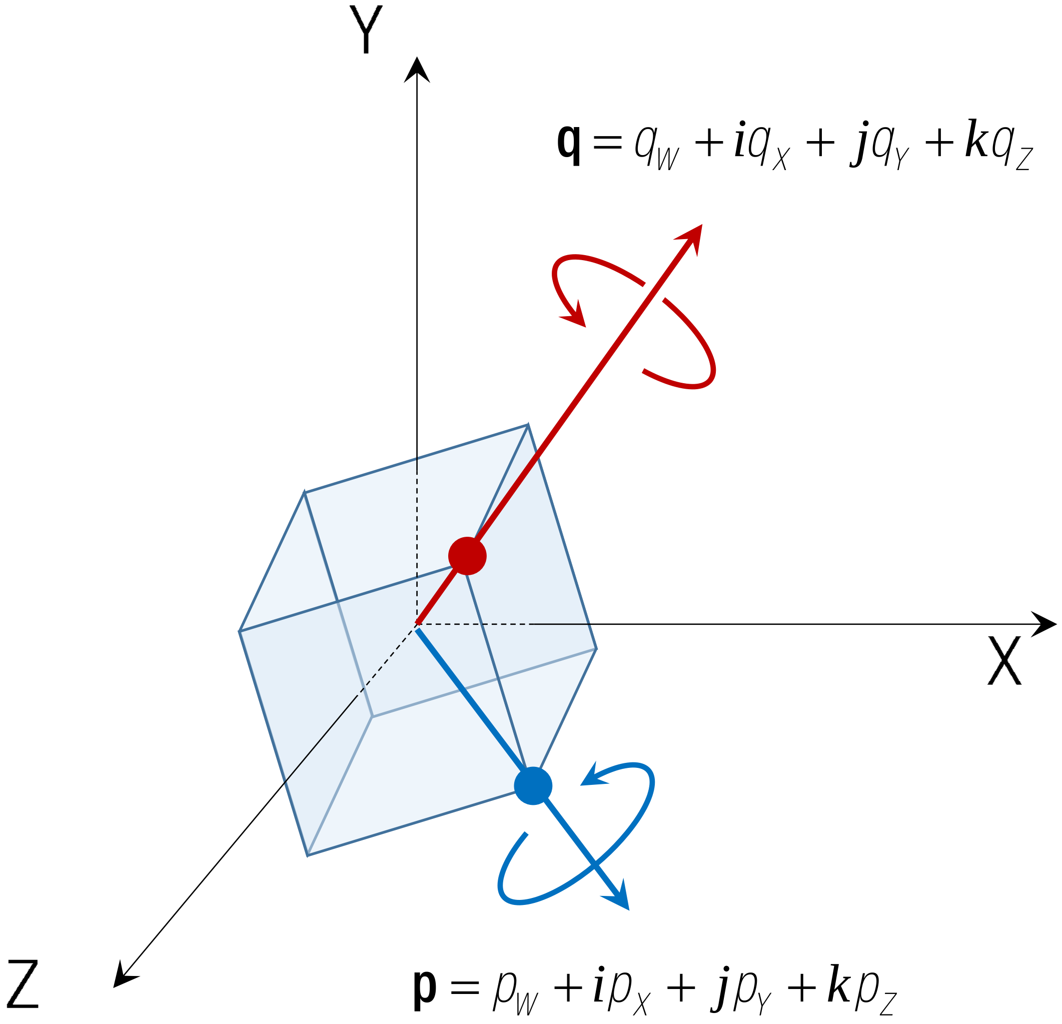
$$\mathbf{e} = \mathbf{a}/\|\mathbf{a}\| = \underbrace{i\frac{3}{5} + j\frac{4}{5} + k\frac{0}{5}}_{\text{Unit vector}}$$

$$\mathbf{q} = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\left(i\frac{3}{5} + j\frac{4}{5} + k\frac{0}{5}\right) \qquad \theta = \frac{\pi}{3}$$

$$= \cos\frac{\pi}{3\cdot 2} + \sin\frac{\pi}{3\cdot 2}\left(i\frac{3}{5} + j\frac{4}{5} + k\frac{0}{5}\right)$$

$$= \frac{\sqrt{3}}{2} + \frac{1}{2}\left(i\frac{3}{5} + j\frac{4}{5} + k\frac{0}{5}\right)$$

Quaternion Product

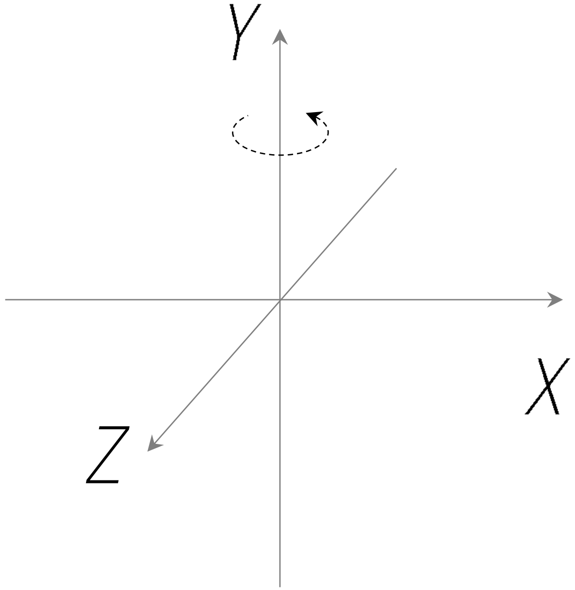


Rotate **q** and then, **p**:

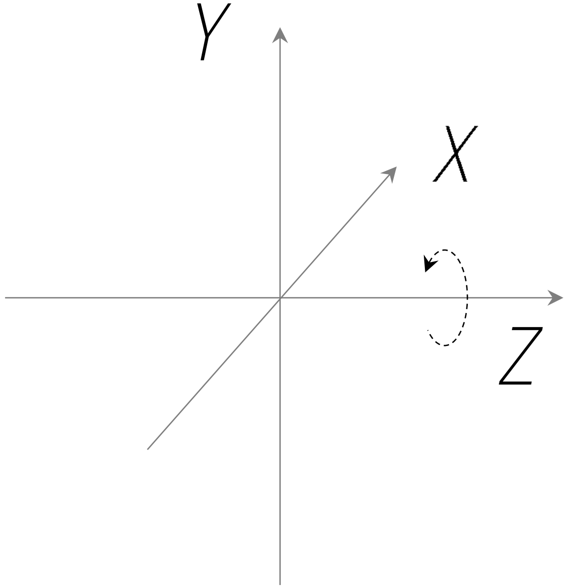
$$
\begin{aligned}
\mathbf{qp} &= \left(q_W + iq_X + jq_Y + kq_Z\right)\left(p_W + ip_X + jp_Y + kp_Z\right) \\
&= \left(q_W p_W - q_X p_X - q_Y p_Y - q_Z p_Z\right) + i\left(q_W p_X + q_X p_W + q_Y p_Z - q_Z p_Y\right) \\
&\quad + j\left(q_W p_Y - q_X p_Z + q_Y p_W + q_Z p_X\right) + k\left(q_W p_Z + q_X p_Y - q_Y p_X + q_Z p_W\right) \\
&= \left(q_W p_W - \hat{\mathbf{q}} \cdot \hat{\mathbf{p}}\right) + \left(q_W \hat{\mathbf{p}} + p_W \hat{\mathbf{q}} + \hat{\mathbf{q}} \times \hat{\mathbf{p}}\right)
\end{aligned}
$$

where $\quad \hat{\mathbf{q}} = iq_X + jq_Y + kq_Z$

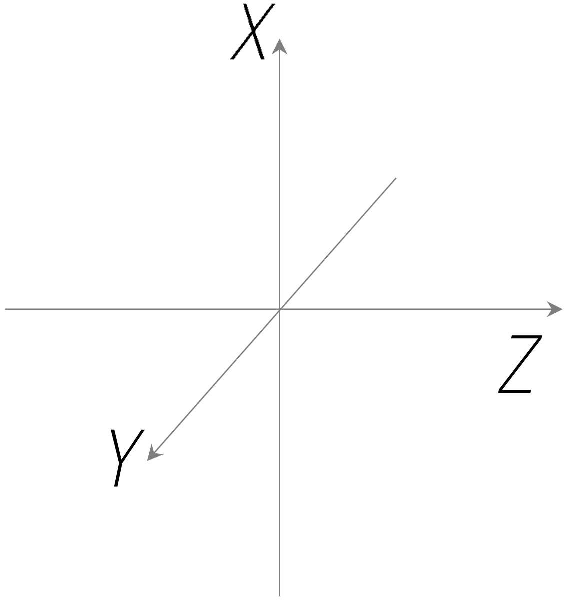
Quaternion Product Example



Rotating 90 degrees about *Y* axis.



Rotating 90 degrees about *Z* axis.



$$\mathbf{q}_1 = \cos\frac{\pi/2}{2} + j\sin\frac{\pi/2}{2}$$

$$\mathbf{q}_2 = \cos\frac{\pi/2}{2} + k\sin\frac{\pi/2}{2}$$

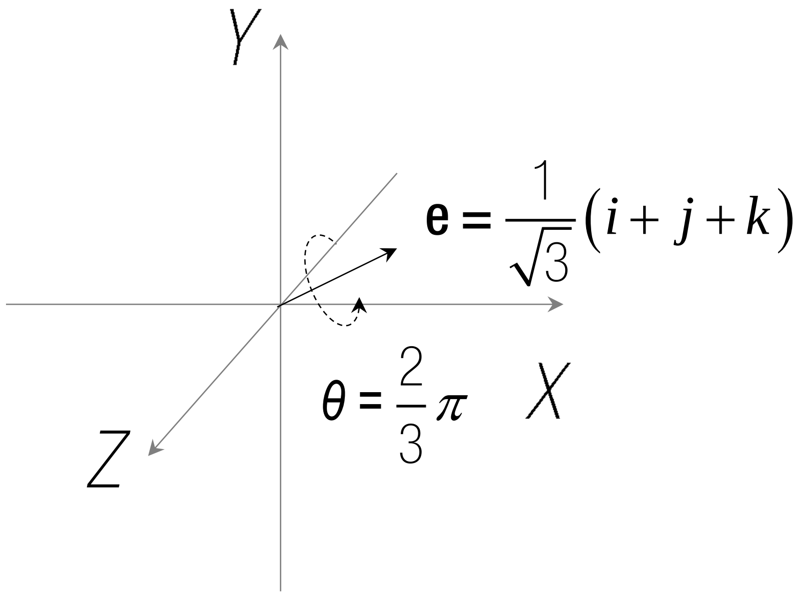
$$\mathbf{q}_{12} = \mathbf{q}_1\mathbf{q}_2 \qquad ?$$

Quaternion Product Example

$$\mathbf{qp} = \left(q_W p_W - \hat{\mathbf{q}} \cdot \hat{\mathbf{p}}\right) + \left(q_W \hat{\mathbf{p}} + p_W \hat{\mathbf{q}} + \hat{\mathbf{q}} \times \hat{\mathbf{p}}\right)$$

$$\mathbf{q}_1 = \cos\frac{\pi/2}{2} + j\sin\frac{\pi/2}{2} \qquad \mathbf{q}_2 = \cos\frac{\pi/2}{2} + k\sin\frac{\pi/2}{2}$$

Quaternion Product Example

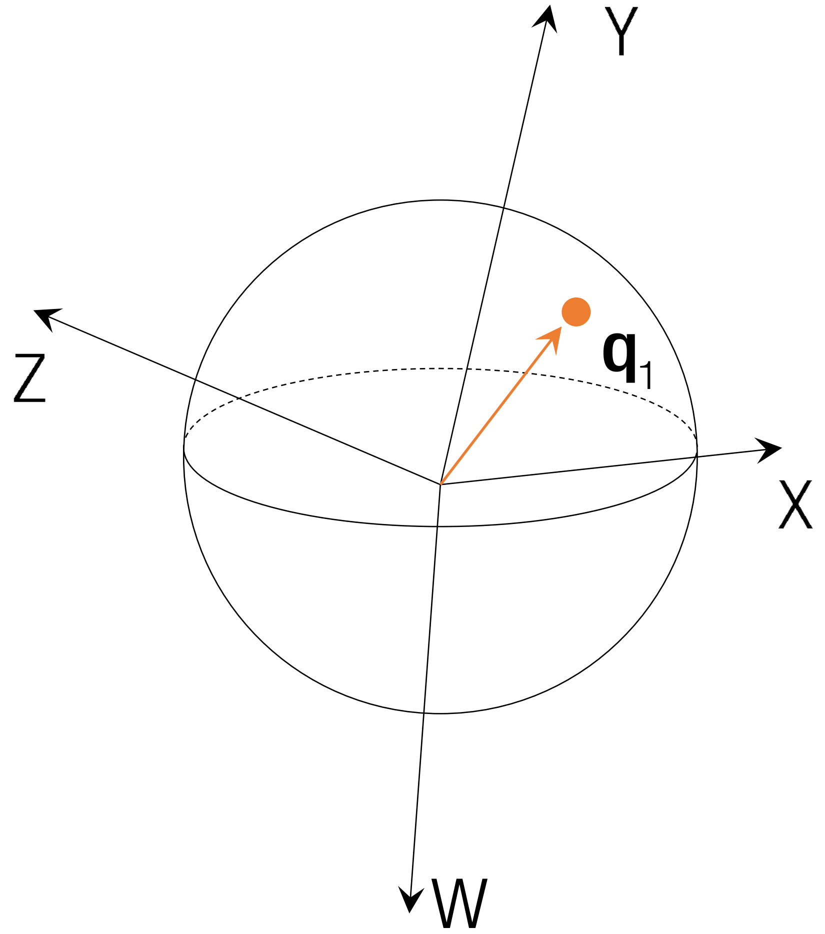


$$\mathbf{qp} = \left(q_W p_W - \hat{\mathbf{q}} \cdot \hat{\mathbf{p}}\right) + \left(q_W \hat{\mathbf{p}} + p_W \hat{\mathbf{q}} + \hat{\mathbf{q}} \times \hat{\mathbf{p}}\right)$$

$$\mathbf{q}_1 = \cos\frac{\pi/2}{2} + j\sin\frac{\pi/2}{2} \qquad \mathbf{q}_2 = \cos\frac{\pi/2}{2} + k\sin\frac{\pi/2}{2}$$

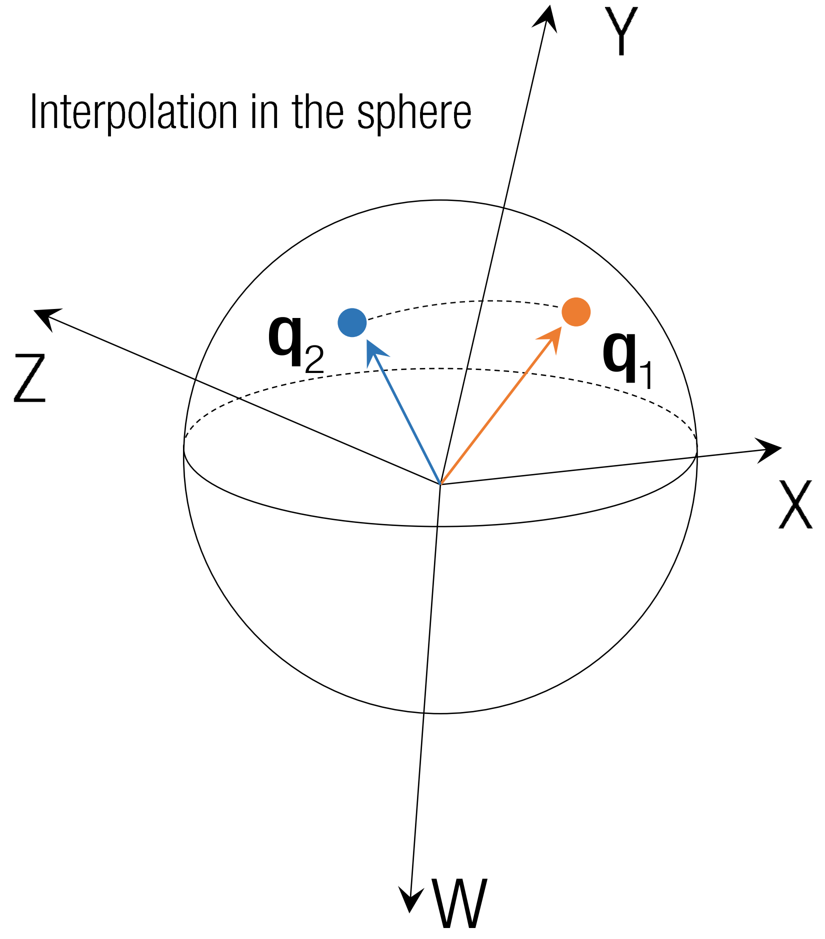
$$\begin{aligned}
\mathbf{q}_{12} &= \mathbf{q}_1\mathbf{q}_2 \\
&= \left(\frac{\sqrt{2}}{2} + \boldsymbol{i}\frac{\sqrt{2}}{2}\right)\left(\frac{\sqrt{2}}{2} + \boldsymbol{j}\frac{\sqrt{2}}{2}\right) \\
&= \frac{1}{2} + \frac{1}{2}\boldsymbol{i} + \frac{1}{2}\boldsymbol{j} + \frac{1}{2}\boldsymbol{k} \\
&= \frac{1}{2} + \frac{1}{2}\boldsymbol{i} + \frac{1}{2}\boldsymbol{j} + \frac{1}{2}\boldsymbol{k} \\
&= \frac{1}{2} + \frac{\sqrt{3}}{2}\left(\frac{1}{\sqrt{3}}\boldsymbol{i} + \frac{1}{\sqrt{3}}\boldsymbol{j} + \frac{1}{\sqrt{3}}\boldsymbol{k}\right)
\end{aligned}$$

Quaternion in 4D Sphere



$$\mathbf{q} = q_W + \boldsymbol{i}q_X + \boldsymbol{j}q_Y + \boldsymbol{k}q_Z = \begin{bmatrix} q_W \\ q_X \\ q_Y \\ q_Z \end{bmatrix} \qquad q_W^2 + q_X^2 + q_Y^2 + q_Z^2 = 1$$

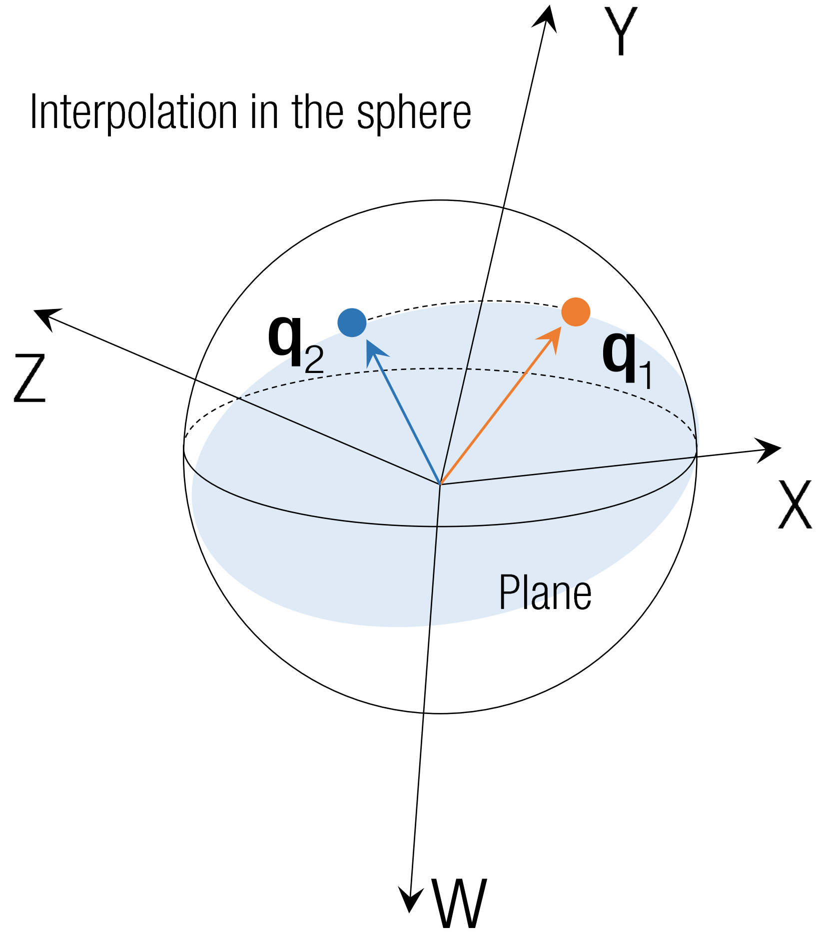
Quaternion in 4D Sphere



$$\mathbf{q} = q_W + iq_X + jq_Y + kq_Z = \begin{bmatrix} q_W \\ q_X \\ q_Y \\ q_Z \end{bmatrix}$$

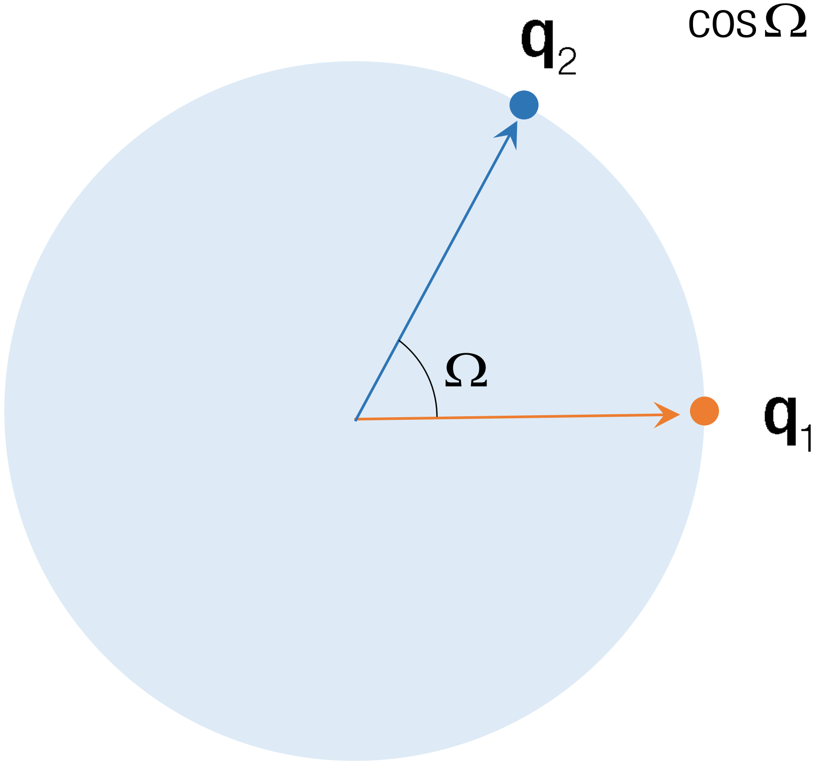
$$q_W^2 + q_X^2 + q_Y^2 + q_Z^2 = 1$$

Quaternion in 4D Sphere

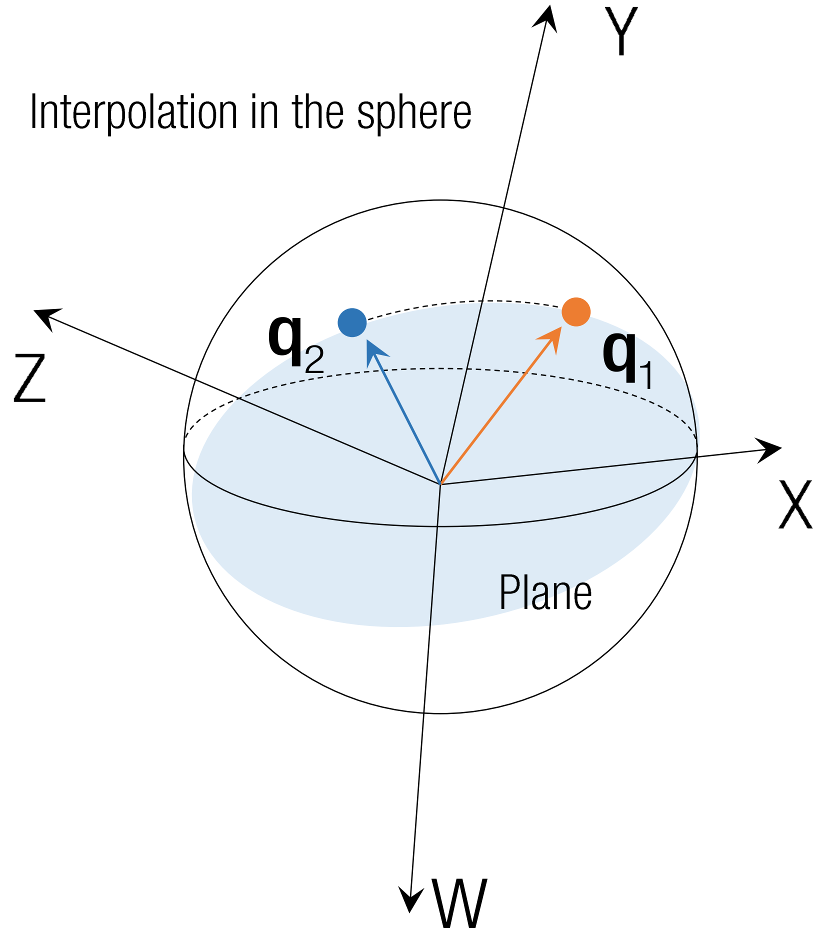


$$\mathbf{q} = q_W + i q_X + j q_Y + k q_Z = \begin{bmatrix} q_W \\ q_X \\ q_Y \\ q_Z \end{bmatrix} \qquad q_W^2 + q_X^2 + q_Y^2 + q_Z^2 = 1$$

$$\cos\Omega = \mathbf{q}_1 \cdot \mathbf{q}_2$$

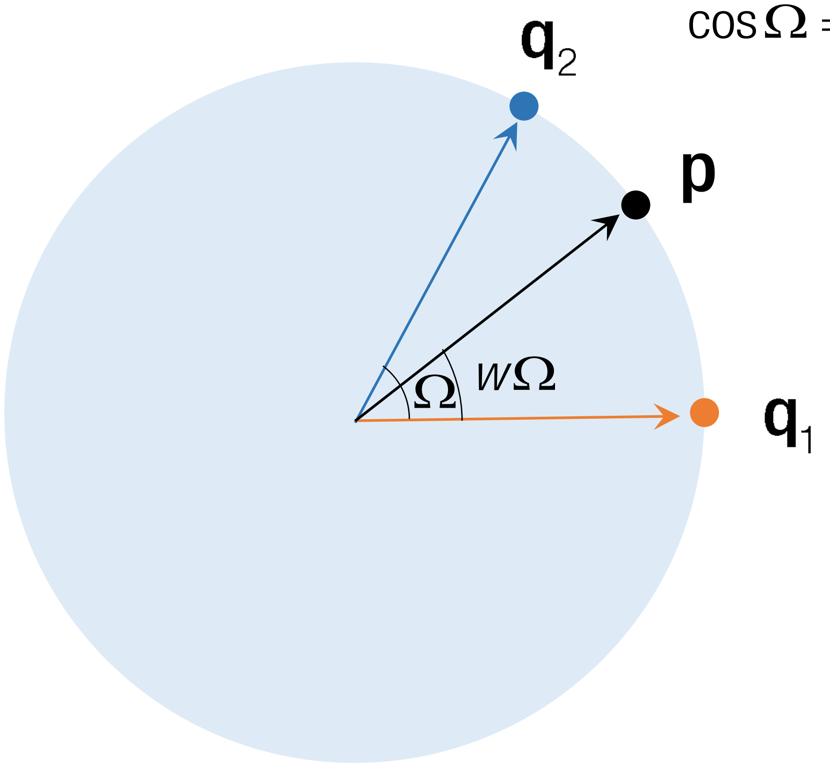


Quaternion Interpolation



$$\mathbf{q} = q_W + iq_X + jq_Y + kq_Z = \begin{bmatrix} q_W \\ q_X \\ q_Y \\ q_Z \end{bmatrix} \qquad q_W^2 + q_X^2 + q_Y^2 + q_Z^2 = 1$$

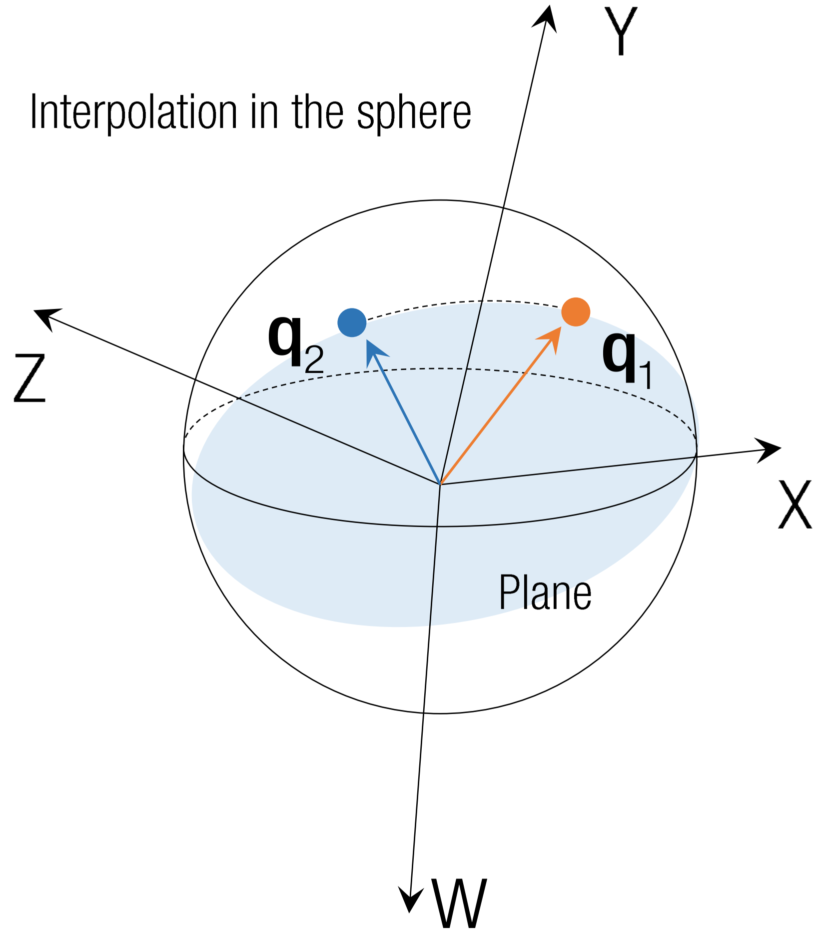
$$\cos\Omega = \mathbf{q}_1 \cdot \mathbf{q}_2$$



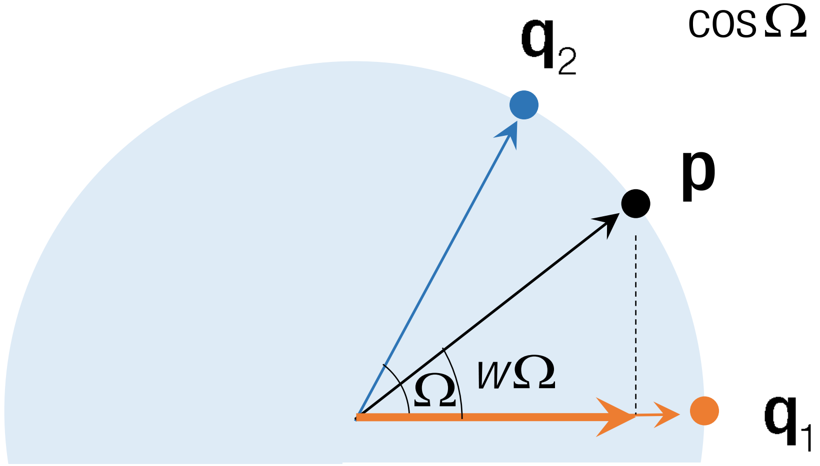
Quaternion Interpolation

$$\mathbf{q} = q_W + iq_X + jq_Y + kq_Z = \begin{bmatrix} q_W \\ q_X \\ q_Y \\ q_Z \end{bmatrix} \qquad q_W^2 + q_X^2 + q_Y^2 + q_Z^2 = 1$$

Interpolation in the sphere

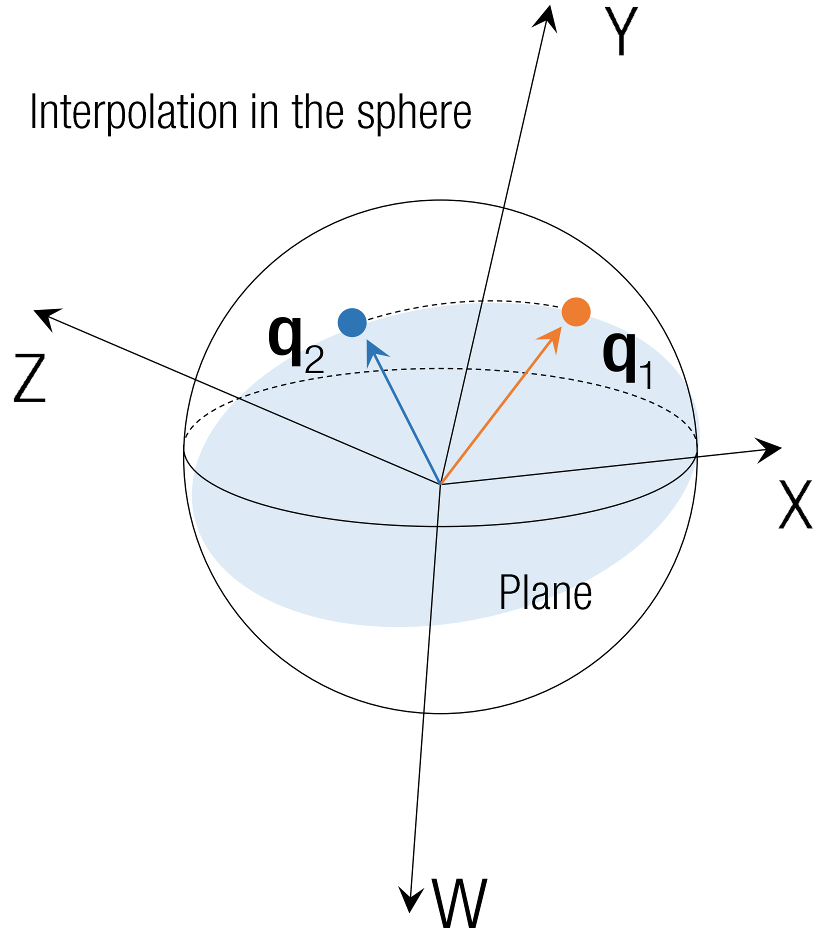


$$\cos\Omega = \mathbf{q}_1 \cdot \mathbf{q}_2$$



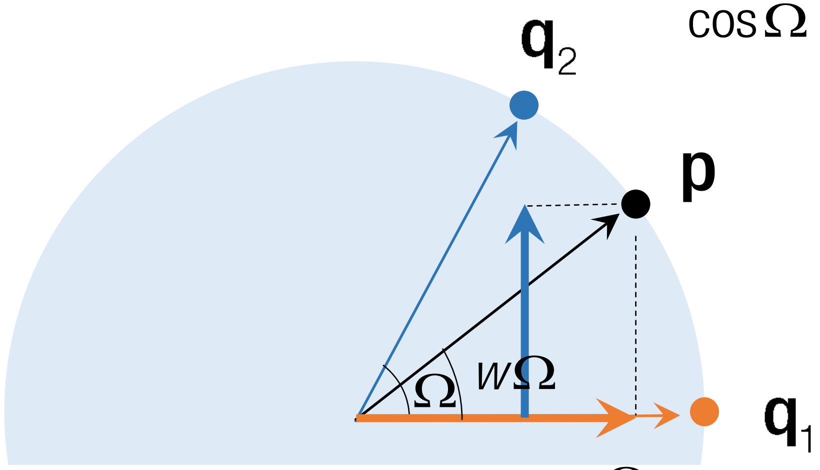
$$\mathbf{p} = \mathbf{q}_1 \cos w\Omega$$

Quaternion Interpolation



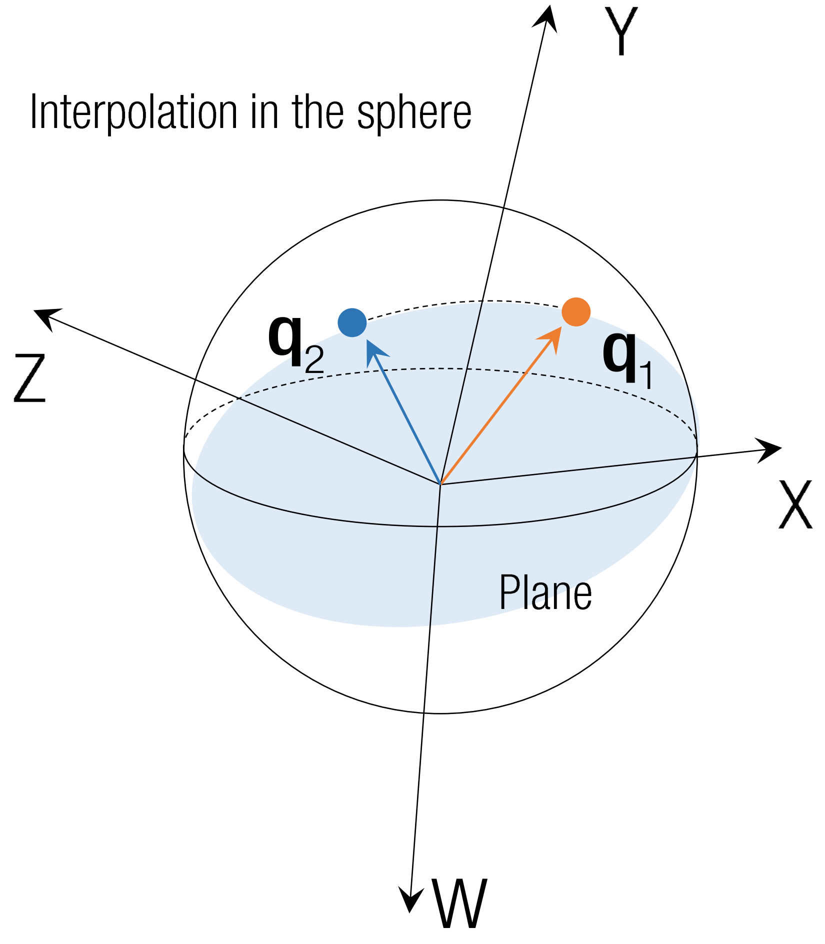
$$\mathbf{q} = q_W + iq_X + jq_Y + kq_Z = \begin{bmatrix} q_W \\ q_X \\ q_Y \\ q_Z \end{bmatrix} \qquad q_W^2 + q_X^2 + q_Y^2 + q_Z^2 = 1$$

$$\cos\Omega = \mathbf{q}_1 \cdot \mathbf{q}_2$$



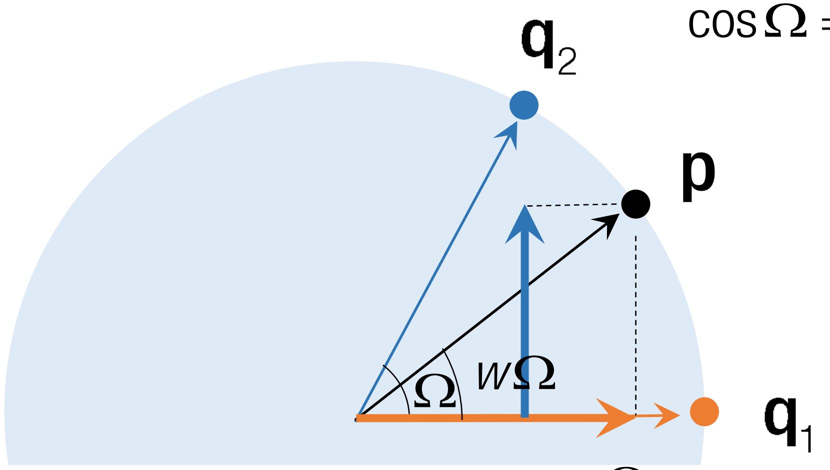
$$\mathbf{p} = \mathbf{q}_1 \cos w\Omega + \frac{\mathbf{q}_2 - \mathbf{q}_1 \cos\Omega}{\sin\Omega} \sin w\Omega$$

Quaternion Interpolation



$$\mathbf{q} = q_W + iq_X + jq_Y + kq_Z = \begin{bmatrix} q_W \\ q_X \\ q_Y \\ q_Z \end{bmatrix} \qquad q_W^2 + q_X^2 + q_Y^2 + q_Z^2 = 1$$

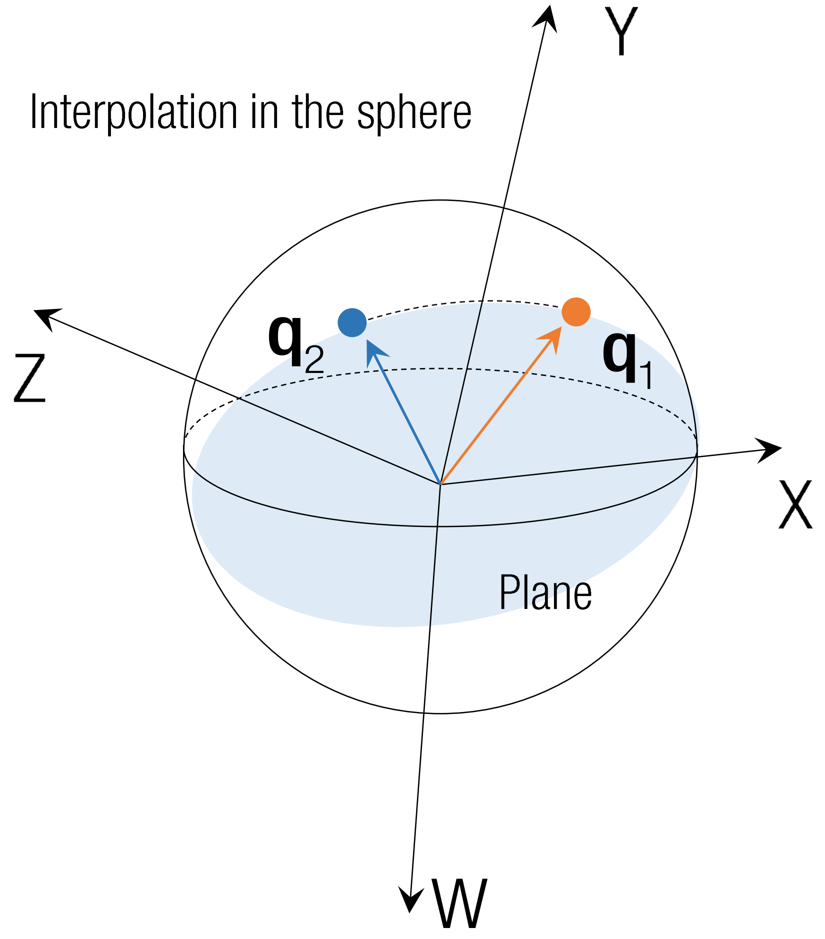
$$\cos\Omega = \mathbf{q}_1 \cdot \mathbf{q}_2$$



$$\mathbf{p} = \mathbf{q}_1 \cos w\Omega + \frac{\mathbf{q}_2 - \mathbf{q}_1 \cos\Omega}{\sin\Omega} \sin w\Omega$$

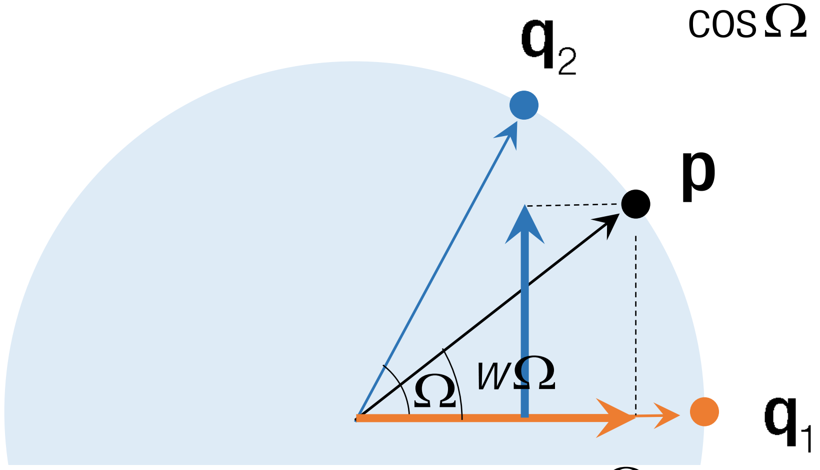
$$= \frac{\mathbf{q}_1 \left(\sin\Omega \cos w\Omega - \cos\Omega \sin w\Omega \right) + \mathbf{q}_2 \sin w\Omega}{\sin\Omega}$$

Quaternion Interpolation



$$\mathbf{q} = q_W + i q_X + j q_Y + k q_Z = \begin{bmatrix} q_W \\ q_X \\ q_Y \\ q_Z \end{bmatrix} \qquad q_W^2 + q_X^2 + q_Y^2 + q_Z^2 = 1$$

$$\cos\Omega = \mathbf{q}_1 \cdot \mathbf{q}_2$$



$$\mathbf{p} = \mathbf{q}_1 \cos w\Omega + \frac{\mathbf{q}_2 - \mathbf{q}_1 \cos\Omega}{\sin\Omega} \sin w\Omega$$

$$= \frac{\mathbf{q}_1 \left(\sin\Omega \cos w\Omega - \cos\Omega \sin w\Omega\right) + \mathbf{q}_2 \sin w\Omega}{\sin\Omega}$$

$$= \frac{\mathbf{q}_1 \sin(1-w)\Omega + \mathbf{q}_2 \sin w\Omega}{\sin\Omega}$$

View Interpolation

Looking left

Looking right

Trajectories with Forensic Analysis
Figure 2: Health status trajectory for varying subpopulations
Figure 3: Shape of the individual quartiles for patients diagnosed with diabetic foot
The instantaneous probability of event in exactly 't' time is defined as
$\lambda_j(t/Z_j) = \lambda_o(t)exp(Z_j\beta)$
The co-efficient vector is estimated through maximizing the partial likelihood
$L(\beta) = \pi_{i:C_i=1} \frac{\theta_i}{\sum_{j:Y_j \geq Y_i}\theta_j}$
Figure 4: Commonly identified trajectories along with their coefficients
Acknowledgements
1UL1RR033183
Department of Computer Science and Engineering
Mayo Clinic, Rochester, MN