UNIVERSITY OF
SOUTH CAROLINA

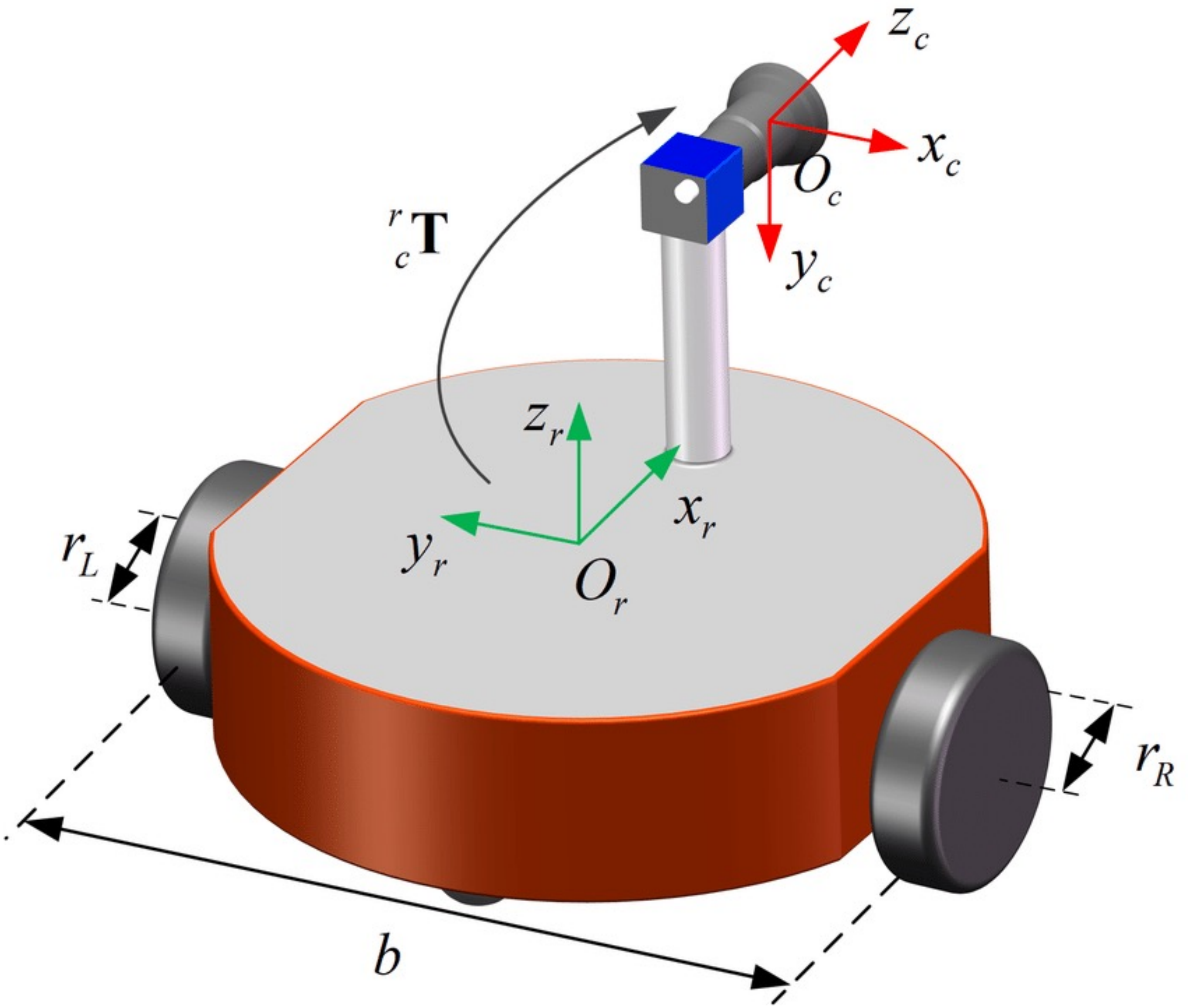
CSCE 574 ROBOTICS

Coordinate Systems

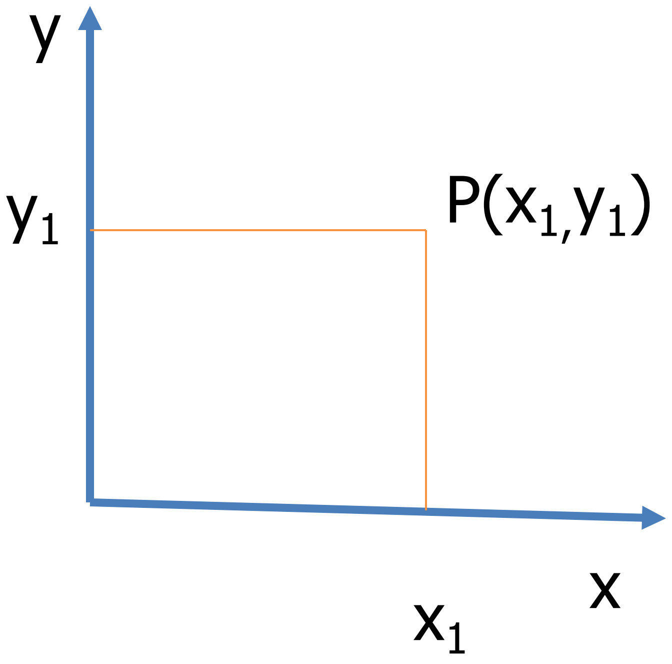
Ioannis Rekleitis

Coordinate Frames

From: https://www.researchgate.net/figure/Schematic-of-a-differential-drive-mobile-robot-equipped-with-a-monocular-camera_fig1_327658659



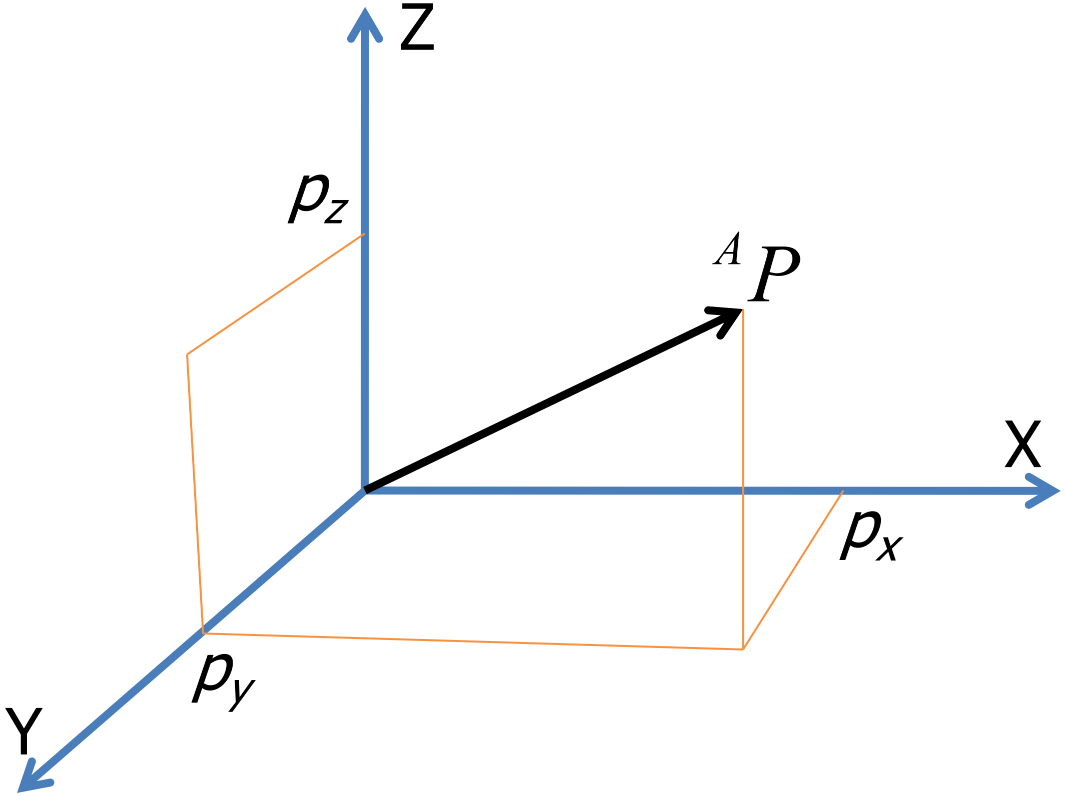
Coordinate Systems



Position Representation

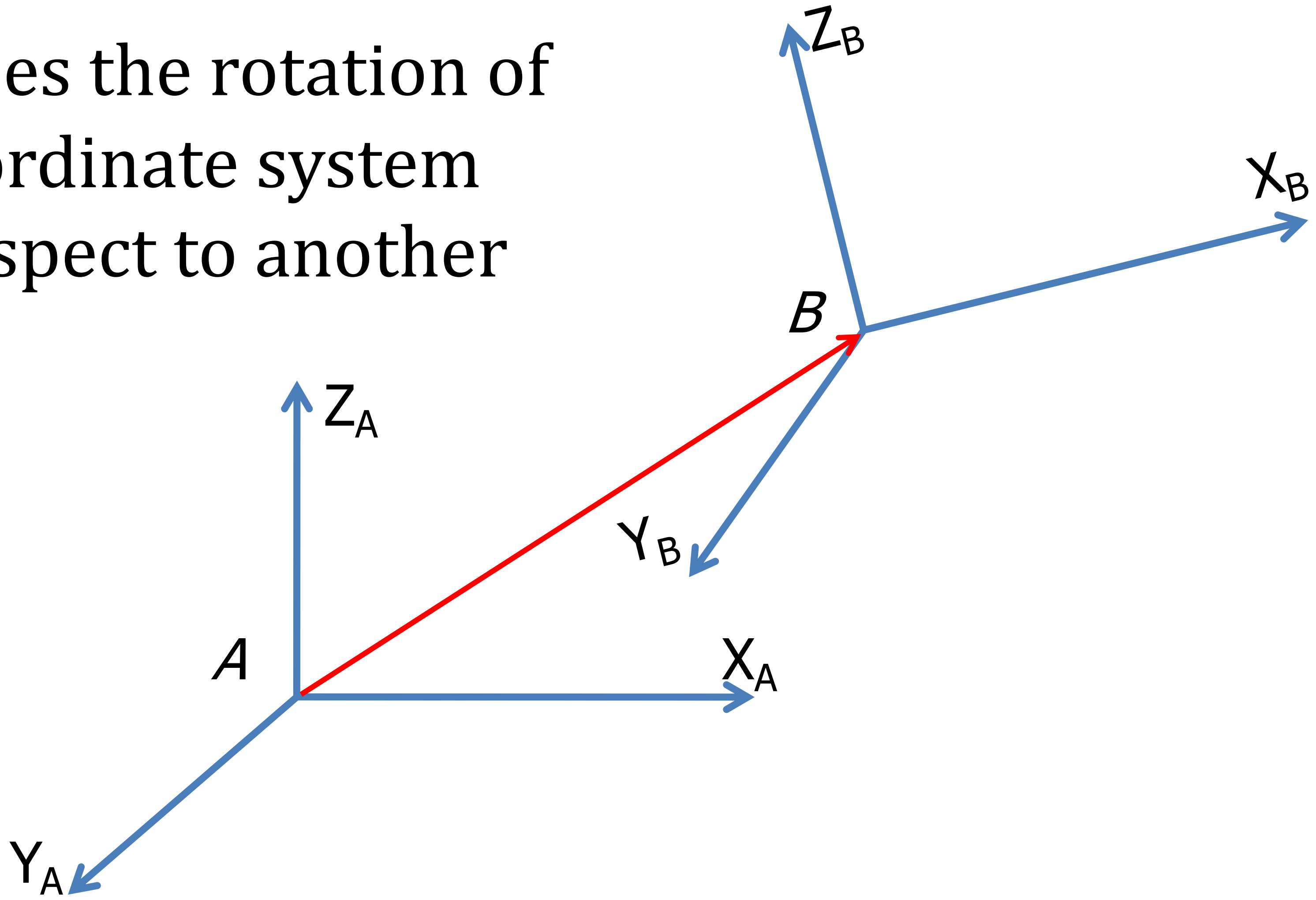
- Position representation is:

$$ {}^{A}P = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} $$



Orientation Representations

- Describes the rotation of one coordinate system with respect to another

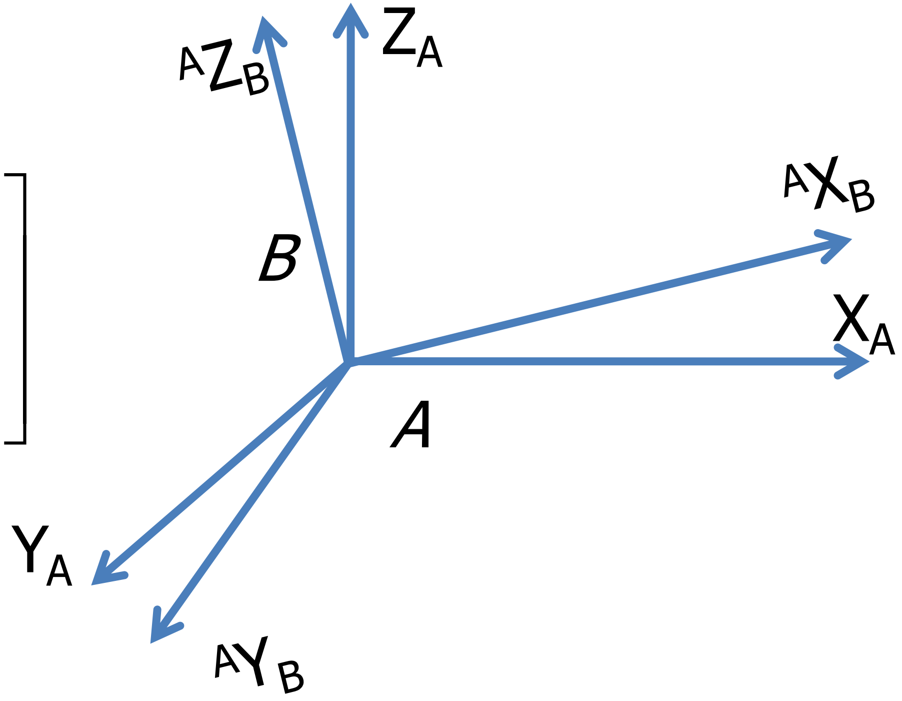


Rotation Matrix

- Write the unit vectors of *B* in the coordinate system of *A*.
- Rotation Matrix:

$$ {}^{A}_{B}R = \begin{bmatrix} {}^{A}\hat{X}_{B} & {}^{A}\hat{Y}_{B} & {}^{A}\hat{Z}_{B} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} $$

$$ = \begin{bmatrix} \hat{X}_{B} \cdot \hat{X}_{A} & \hat{Y}_{B} \cdot \hat{X}_{A} & \hat{Z}_{B} \cdot \hat{X}_{A} \\ \hat{X}_{B} \cdot \hat{Y}_{A} & \hat{Y}_{B} \cdot \hat{Y}_{A} & \hat{Z}_{B} \cdot \hat{Y}_{A} \\ \hat{X}_{B} \cdot \hat{Z}_{A} & \hat{Y}_{B} \cdot \hat{Z}_{A} & \hat{Z}_{B} \cdot \hat{Z}_{A} \end{bmatrix} $$



Properties of Rotation Matrix

$$ {}^{B}_{A}R = {}^{A}_{B}R^{T} $$

$$ {}^{A}_{B}R^{T}\,{}^{A}_{B}R = I_{3} $$

$$ {}^{A}_{B}R = {}^{B}_{A}R^{-1} = {}^{B}_{A}R^{T} $$

Coordinate System Transformation

$$M = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0_{3\times1} & 1 \end{bmatrix}$$

where R is the rotation matrix and T is the translation vector

Rotation Matrix

- The rotation matrix consists of 9 variables, but there are many constraints. The minimum number of variables needed to describe a rotation is three.

Rotation Matrix-Single Axis

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

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

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

Fixed Angles

- One simple method is to perform three rotations about the axis of the original coordinate frame:
  - X-Y-Z fixed angles

$$
{}^{A}_{B}R(\theta,\phi,\psi)=R_z(\psi)R_y(\phi)R_x(\theta)
$$

$$
=\begin{bmatrix}
\cos(\psi)\cos(\phi) & \cos(\psi)\sin(\phi)\sin(\theta)-\sin(\psi)\cos(\theta) & \cos(\psi)\sin(\phi)\cos(\theta)+\sin(\psi)\sin(\theta) \\
\sin(\psi)\cos(\phi) & \sin(\psi)\sin(\phi)\sin(\theta)+\cos(\psi)\cos(\theta) & \sin(\psi)\sin(\phi)\cos(\theta)+\cos(\psi)\sin(\theta) \\
-\sin(\phi) & \cos(\phi)\sin(\theta) & \cos(\theta)\cos(\psi)
\end{bmatrix}
$$

- There are 12 different combinations

Inverse Problem

- From a Rotation matrix find the fixed angle rotations:

$$
{}^{A}_{B}R(\theta,\phi,\psi) = {}^{A}_{B}R \Rightarrow
$$

$$
\begin{bmatrix}
\cos(\psi)\cos(\phi) & \cos(\psi)\sin(\phi)\sin(\theta)-\sin(\psi)\cos(\theta) & \cos(\psi)\sin(\phi)\cos(\theta)+\sin(\psi)\sin(\theta) \\
\sin(\psi)\cos(\phi) & \sin(\psi)\sin(\phi)\sin(\theta)+\cos(\psi)\cos(\theta) & \sin(\psi)\sin(\phi)\cos(\theta)+\cos(\psi)\sin(\theta) \\
-\sin(\phi) & \cos(\phi)\sin(\theta) & \cos(\theta)\cos(\psi)
\end{bmatrix}
=
\begin{bmatrix}
r_{11} & r_{12} & r_{13} \\
r_{21} & r_{22} & r_{23} \\
r_{31} & r_{32} & r_{33}
\end{bmatrix}
$$

thus :

$$
\phi = A\tan 2\left(-r_{31}, \sqrt{\left(r_{11}^{2} + r_{21}^{2}\right)}\right)
$$

$$
\psi = A\tan 2\left({}^{r_{21}}\!/_{\cos(\phi)}, {}^{r_{11}}\!/_{\cos(\phi)} \right)
$$

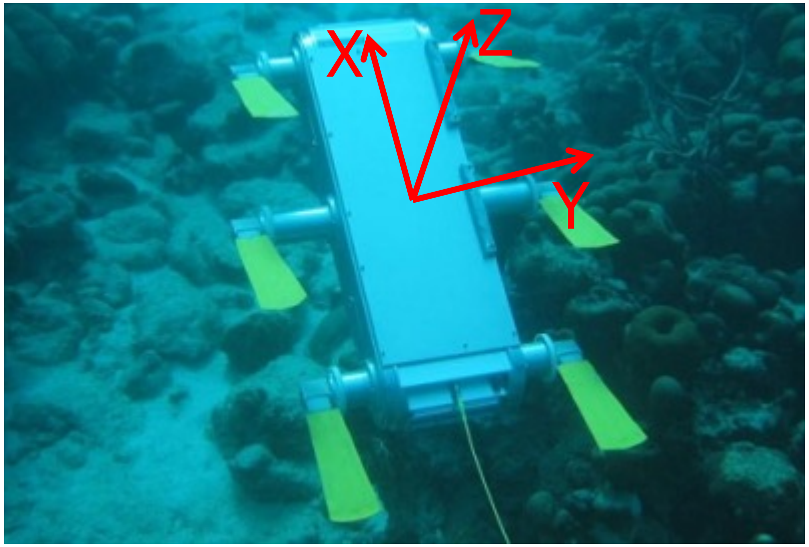
$$
\theta = A\tan 2\left({}^{r_{32}}\!/_{\cos(\phi)}, {}^{r_{33}}\!/_{\cos(\phi)} \right)
$$

Euler Angles

- **ZYX**: Starting with the two frames aligned, first rotate about the Z_B axis, then by the Y_B axis and then by the X_B axis. The results are the same as with using XYZ fixed angle rotation.
- There are 12 different combination of Euler Angle representations

Euler Angles

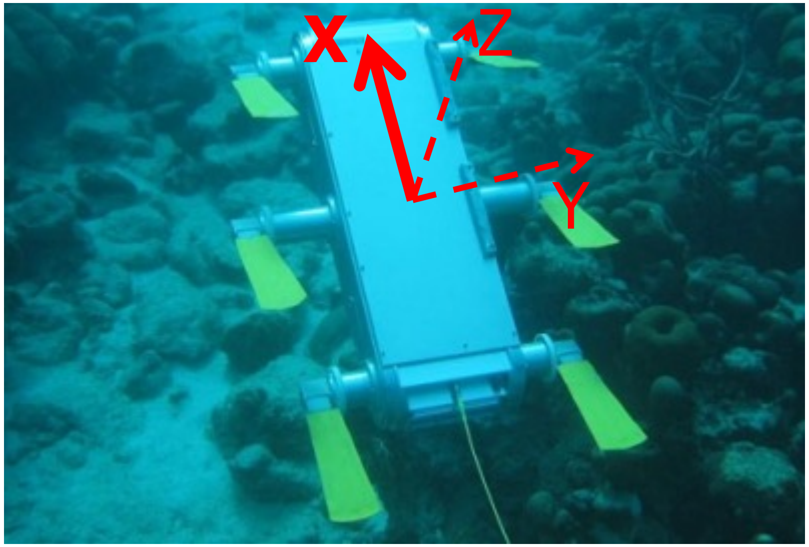
- Traditionally the three angles along the axis are called Roll, Pitch, and Yaw



Euler Angles

- Traditionally the three angles along the axis are called Roll, Pitch, and Yaw

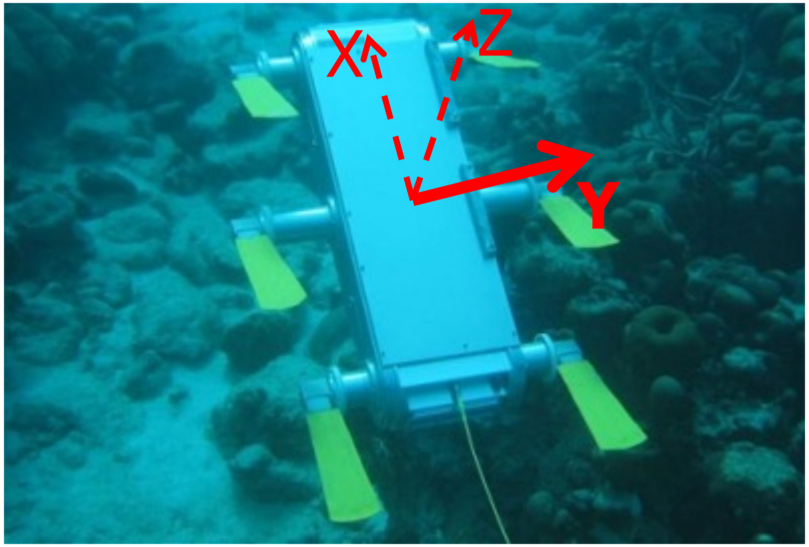
Roll



Euler Angles

- Traditionally the three angles along the axis are called Roll, Pitch, and Yaw

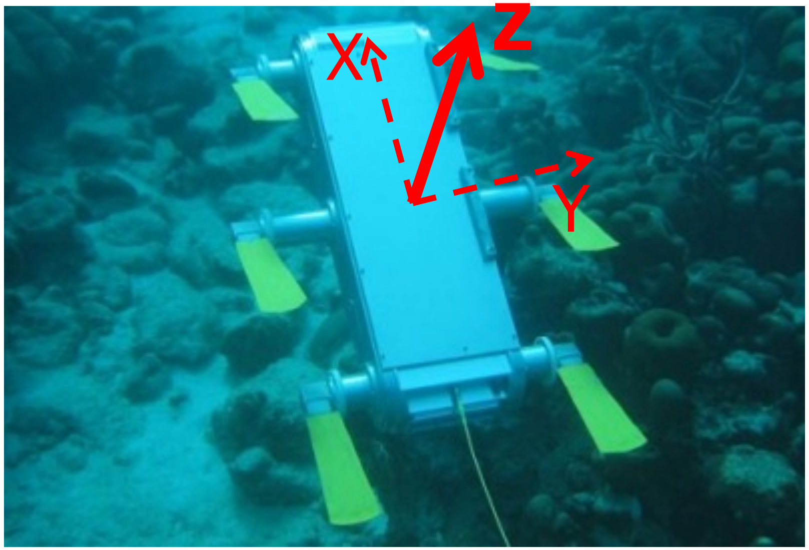
Pitch



Euler Angles

- Traditionally the three angles along the axis are called Roll, Pitch, and Yaw

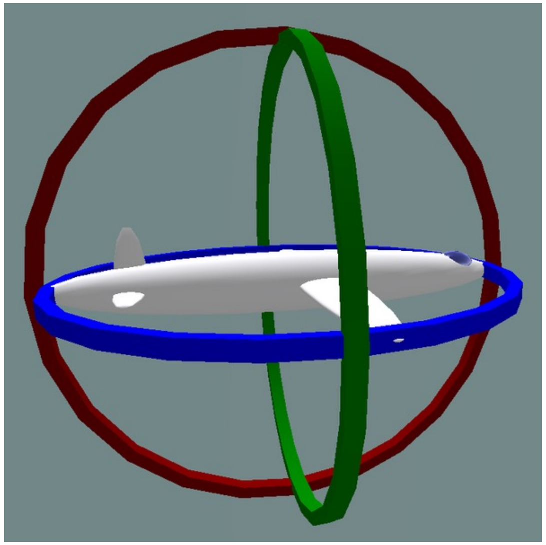
Yaw

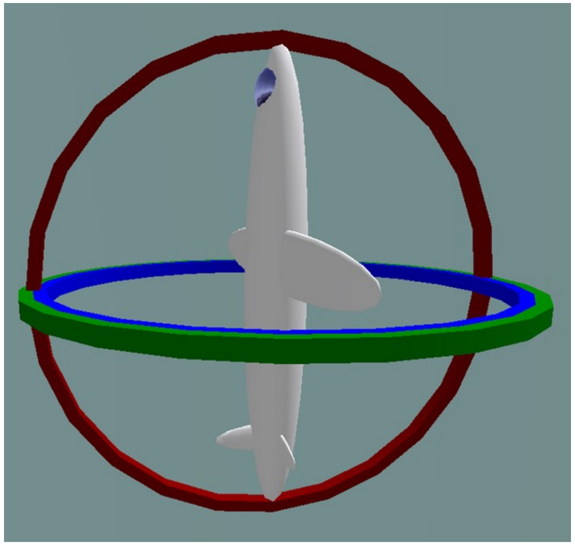


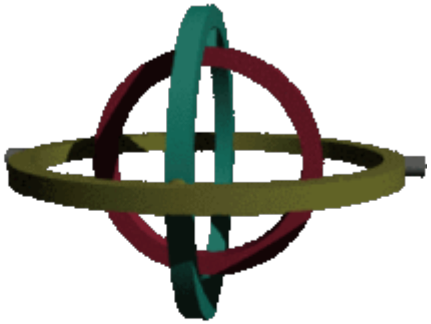
Euler Angle concerns: Gimbal Lock

Using the **ZYZ** convention

- (90°, 45°, −105°) ≡ (−270°, −315°, 255°) multiples of 360°
- (72°, 0°, 0°) ≡ (40°, 0°, 32°) singular alignment (Gimbal lock)
- (45°, 60°, −30°) ≡ (−135°, −60°, 150°) bistable flip

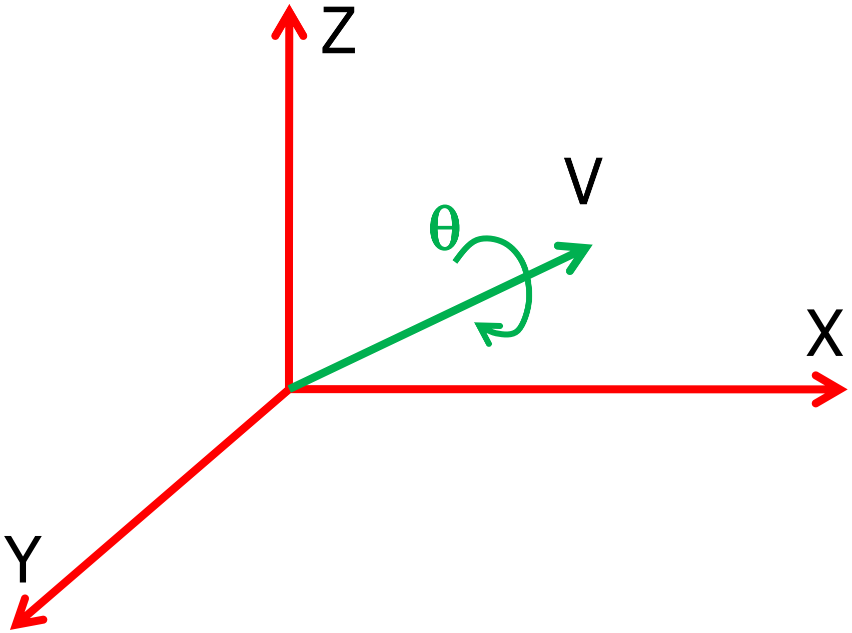






Axis-Angle Representation

- Represent an arbitrary rotation as a combination of a vector and an angle



Quaternions

- Are similar to axis-angle representation
- Two formulations
  - Classical
  - Based on JPL's standards

    W. G. Breckenridge, "Quaternions - Proposed Standard Conventions," JPL, Tech. Rep. INTEROFFICE MEMORANDUM IOM 343-79-1199, 1999.
- Avoids Gimbal lock

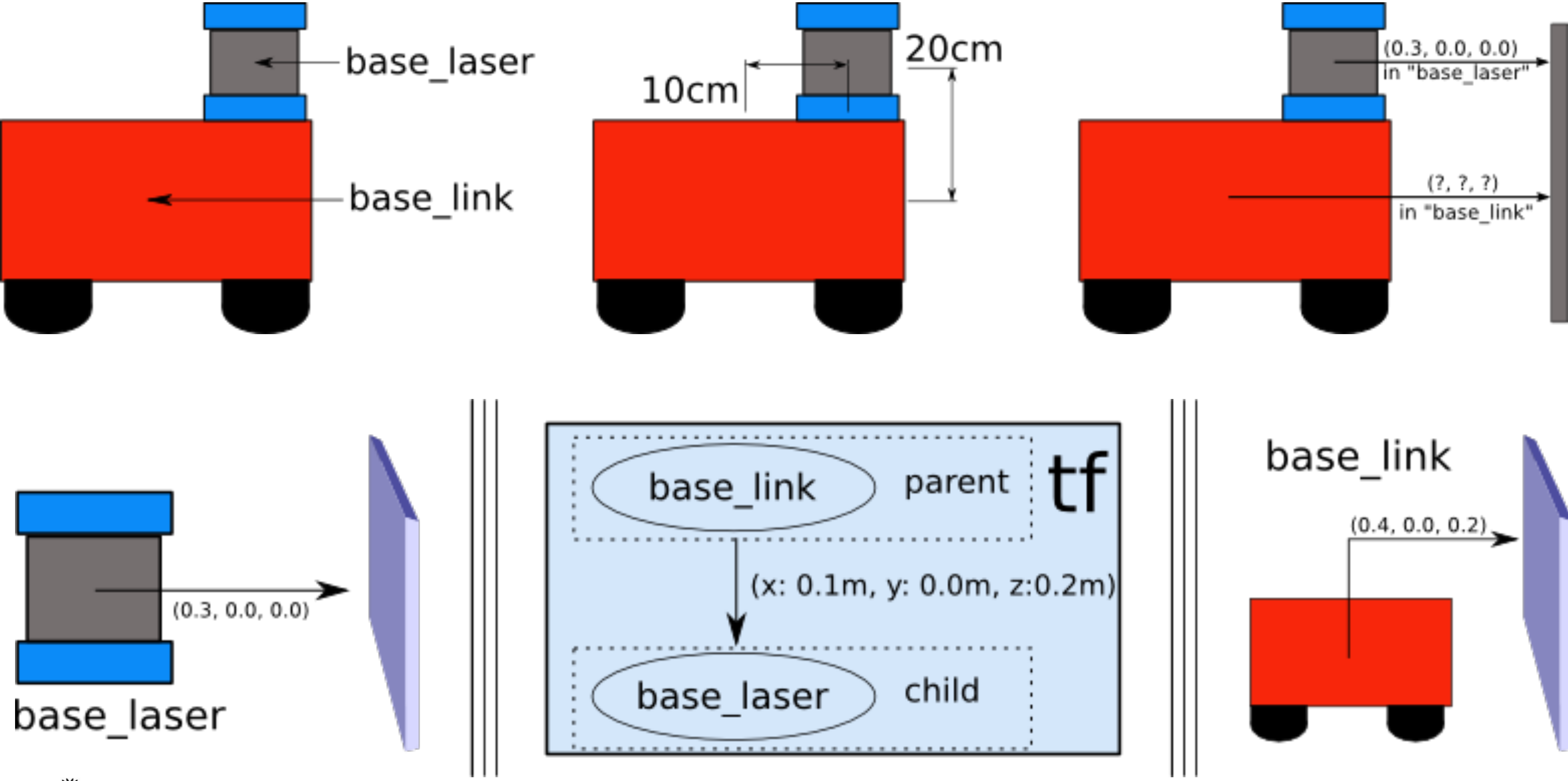
- See also: M. D. Shuster, "A survey of attitude representations," Journal of the Astronautical Sciences, vol. 41, no. 4, pp. 439–517, Oct.–Dec. 1993.

Quaternions

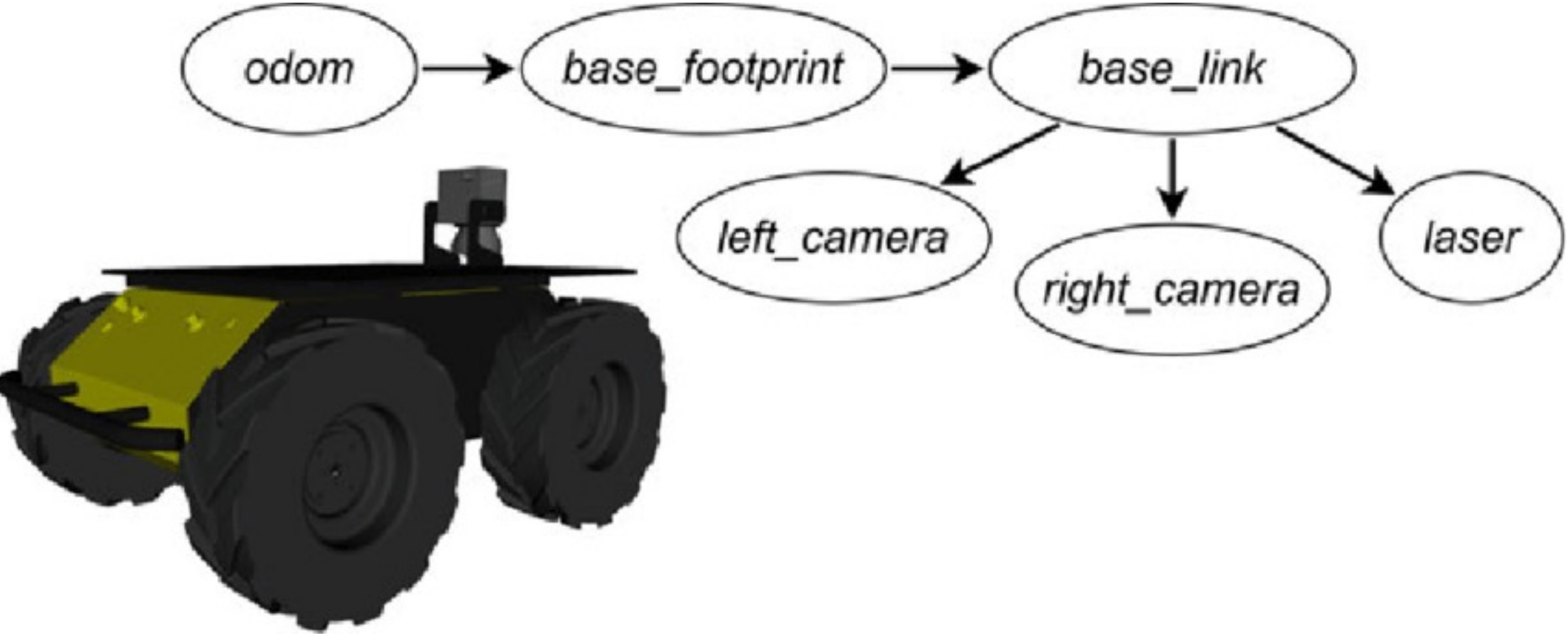
	Classic notation	JPL-based
	$\bar{q} = q_4 + q_1 i + q_2 j + q_3 k$	$\bar{q} = q_4 + q_1 i + q_2 j + q_3 k$
	$i^2 = j^2 = k^2 = ijk = -1$ $ij = -ji = k,\ jk = -kj = i,\ ki = -ik = j$	$i^2 = j^2 = k^2 = -1$ $-ij = ji = k, -jk = kj = i, -ki = ik = j$
Vector Notation	$\bar{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}, q_0 = \cos\left(\theta/2\right), \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \sin\left(\theta/2\right)\cos(\beta_x) \\ \sin\left(\theta/2\right)\cos(\beta_y) \\ \sin\left(\theta/2\right)\cos(\beta_z) \end{bmatrix}$	$\bar{q} = \begin{bmatrix} \mathbf{q} \\ q_4 \end{bmatrix}, \mathbf{q} = \begin{bmatrix} k_x \sin\left(\theta/2\right) \\ k_y \sin\left(\theta/2\right) \\ k_z \sin\left(\theta/2\right) \end{bmatrix}, q_4 = \cos\left(\theta/2\right)$
		$\lVert\bar{q}\rVert = 1, \bar{q} \otimes \bar{p}, \mathbf{q} \times \mathbf{p}, \bar{q}_I, \lfloor \mathbf{q} \times \rfloor$

See also: N. Trawny and S. I. Roumeliotis, “Indirect Kalman Filter for 3D Attitude Estimation,” University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep. 2005-002, March 2005.

Use ROS tf package



Coordinate frames



Coordinate frames on PR2