

Practical Course: Vision-based Navigation Winter Semester 2019

Lecture 1. 3D Geometry and Lie Groups

Vladyslav Usenko, Nikolaus Demmel,
Prof. Dr. Daniel Cremers

Contents

- Course contents and preliminary knowledge
- Framework and mathematic form of a SLAM problem
- 3D geometry
- Lie groups

Contents

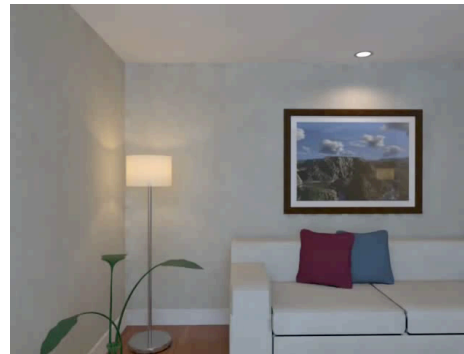
- Course contents and preliminary knowledge
- Framework and mathematic form of a SLAM problem
- 3D geometry
- Lie groups

1. Course contents and preliminary knowledge

- General overview of computer vision tasks

1. Course contents and preliminary knowledge

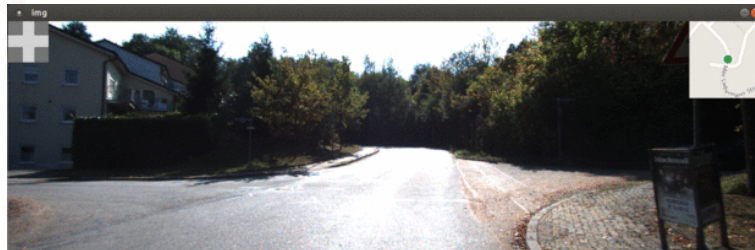
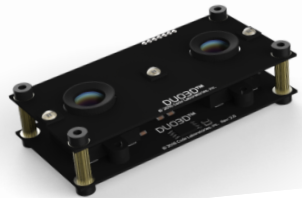
- Computer vision



Object detection
Object recognition
Object tracking
Segmentation

...

SLAM



Real world cameras

Image and video sequences

CV tasks

1. Course contents and preliminary knowledge

- What is SLAM? Simultaneous localization and mapping



Universidad
Zaragoza



Instituto Universitario de Investigación
en Ingeniería de Aragón
Universidad Zaragoza

ORB-SLAM2: an Open-Source SLAM System
for Monocular, Stereo and RGB-D Cameras

Raúl Mur-Artal and Juan D. Tardós

raulmur@unizar.es

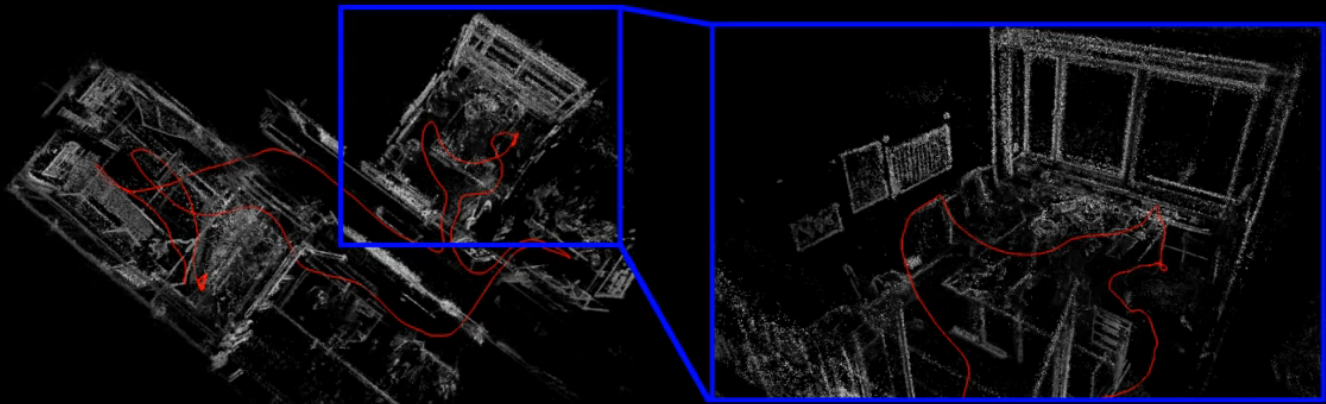
tardos@unizar.es

Indoor/outdoor localization


1. Course contents and preliminary knowledge

- What is SLAM?

Direct Sparse Odometry
Jakob Engel,^{1,2} Vladlen Koltun,² Daniel Cremers¹
July 2016



TUM ¹Computer Vision Group
Technical University Munich

²Intel Labs 

Dense/semi-dense reconstruction

1. Course contents and preliminary knowledge

- What is SLAM?

ElasticFusion: Dense SLAM Without A Pose Graph

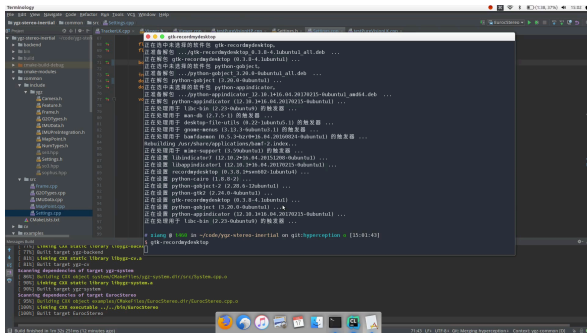
Thomas Whelan, Stefan Leutenegger, Renato Salas-Moreno, Ben Glocker, Andrew Davison

Imperial College London

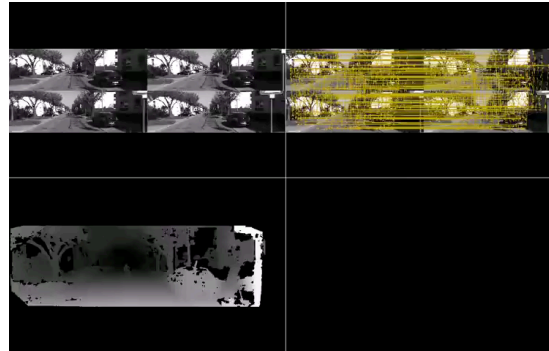
RGB-D dense reconstruction

1. Course contents and preliminary knowledge

- SLAM applications



Hand-held devices



Autonomous driving

De-noising, Stabilizing and Completing 3D Reconstructions On-the-go using Plane Priors

Maksym Dzitsiuk, Jürgen Sturm, Robert Maier, Lingni Ma, Daniel Cremers



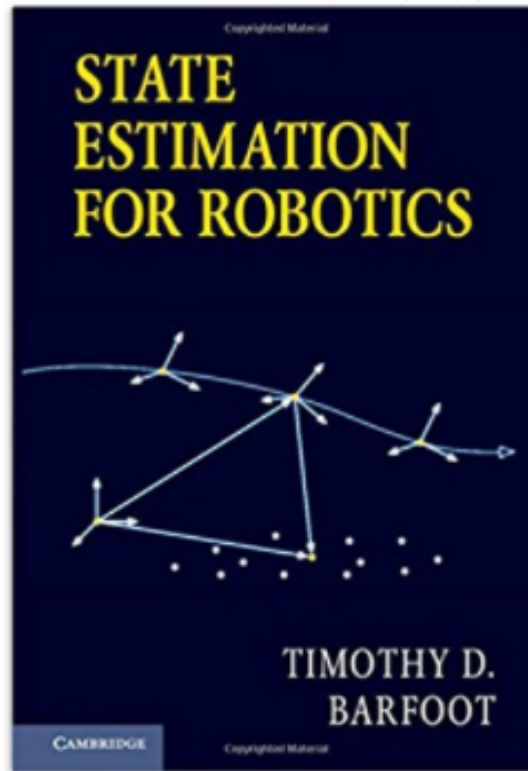
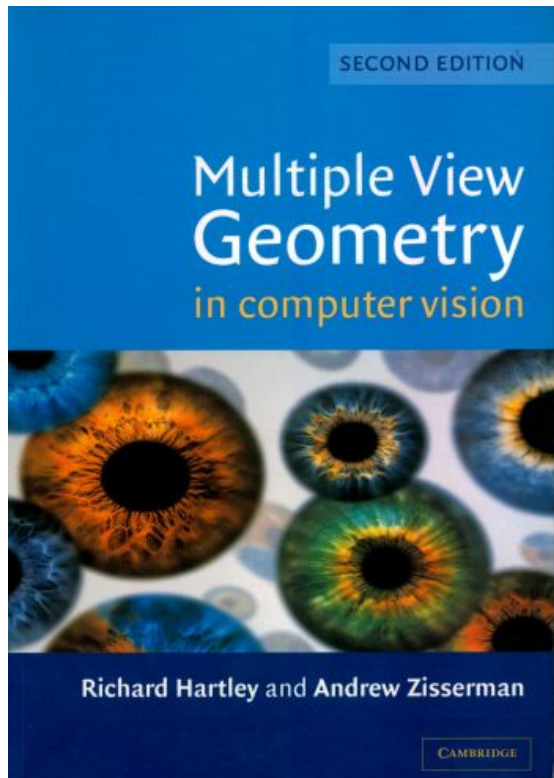
Technical University of Munich,
Google



Augmented reality/VR

1. Course contents and preliminary knowledge

- Computer vision



Harley and Zisserman,
Multiple view geometry
in computer vision

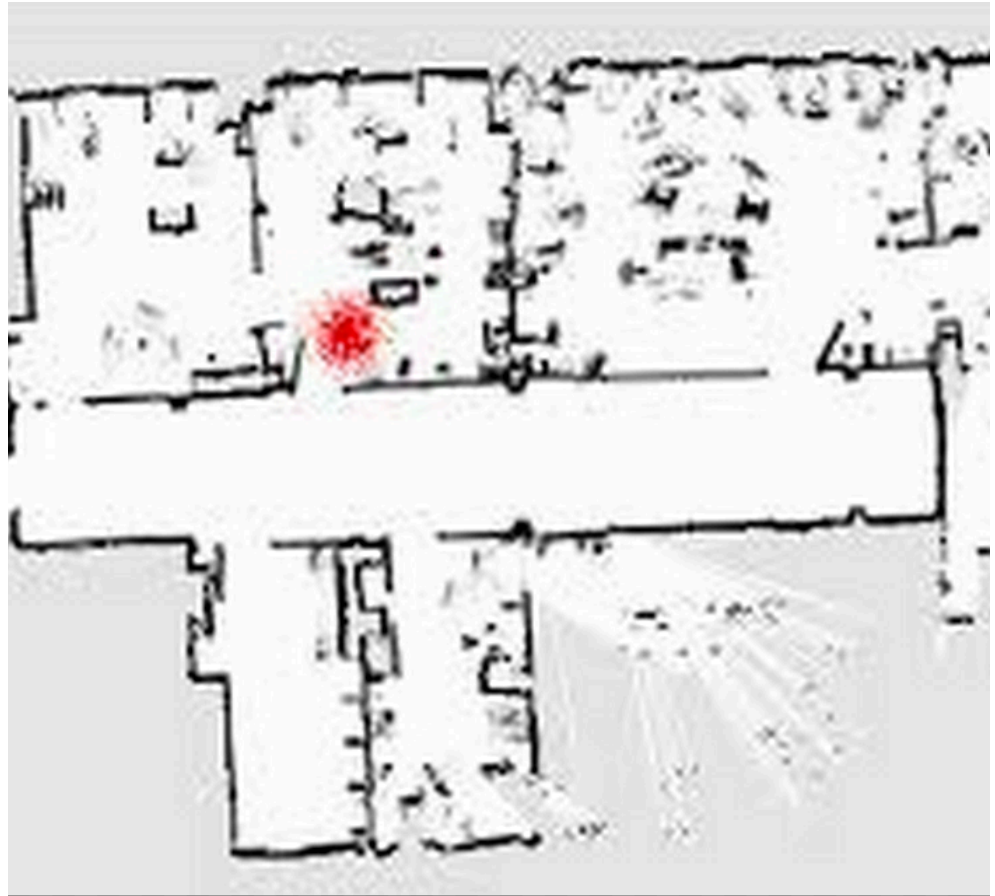
Tim Barfoot, State
estimation for robotics

Contents

- Course contents and preliminary knowledge
- Framework and mathematic form of a SLAM problem
- 3D geometry
- Lie groups

2. Framework of SLAM

- SLAM problem
 - Fundamental problems in intelligent robots
 - Where am I?
 - Localization
 - What is around me?
 - Mapping
- Chicken and egg problem
 - Localization needs accurate map
 - Mapping needs accurate localization



2. Framework of SLAM

- How to do SLAM? -Sensors
- Sensor is the way to measure the outside environment
- Interoceptive sensors: accelerometer, gyroscope ...
- Exteroceptive sensors: camera, laser rangefinder, GPS ...



(a)



(b)



(c)



(d)



(e)



(f)

Some sensors must be placed in a cooperative environment, other can be directly equipped in the robot itself

2. Framework of SLAM

- Visual SLAM
- Cameras
 - Monocular
 - Stereo
 - RGB-D
 - Omnidirectional, Event camera, etc

- Cameras
 - Cheap, rich information
 - Record 2D projected image of the 3D world
 - The 3D-2D projection throws away one dimension: distance



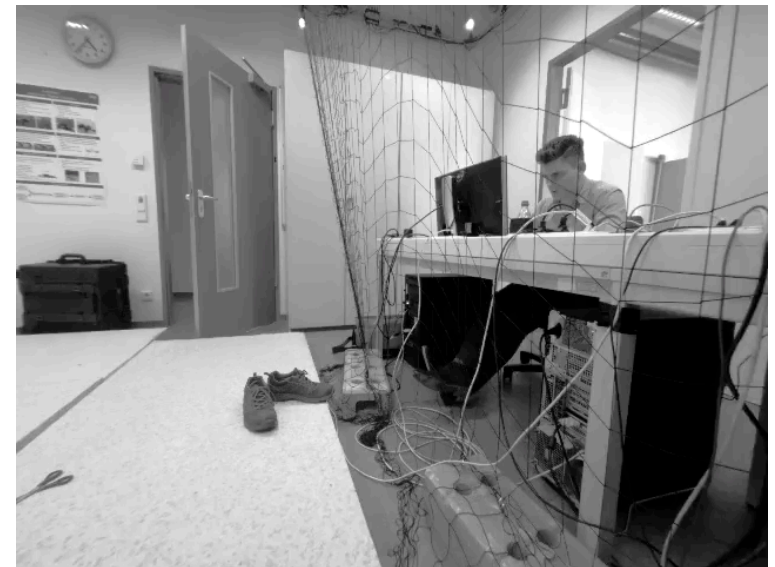
Monocular camera



RGB-D (depth) camera

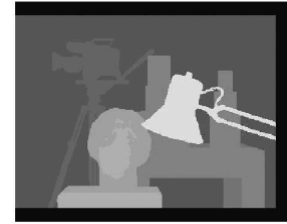
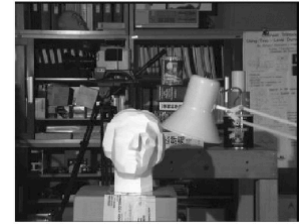
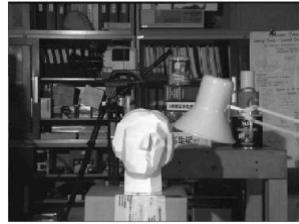


Stereo camera

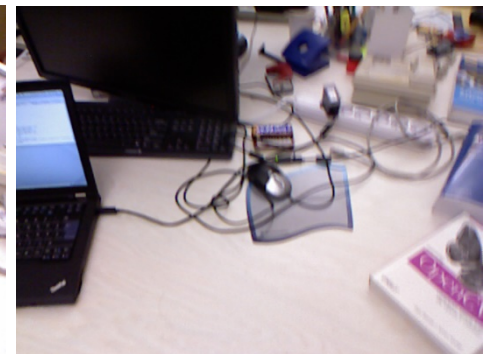


2. Framework of SLAM

- Various kinds of cameras:
- Monocular: image only, need other methods to estimate the depth
- Stereo: disparity to depth
- RGB-D: physical depth measurements



Stereo vision estimates the depth from disparity



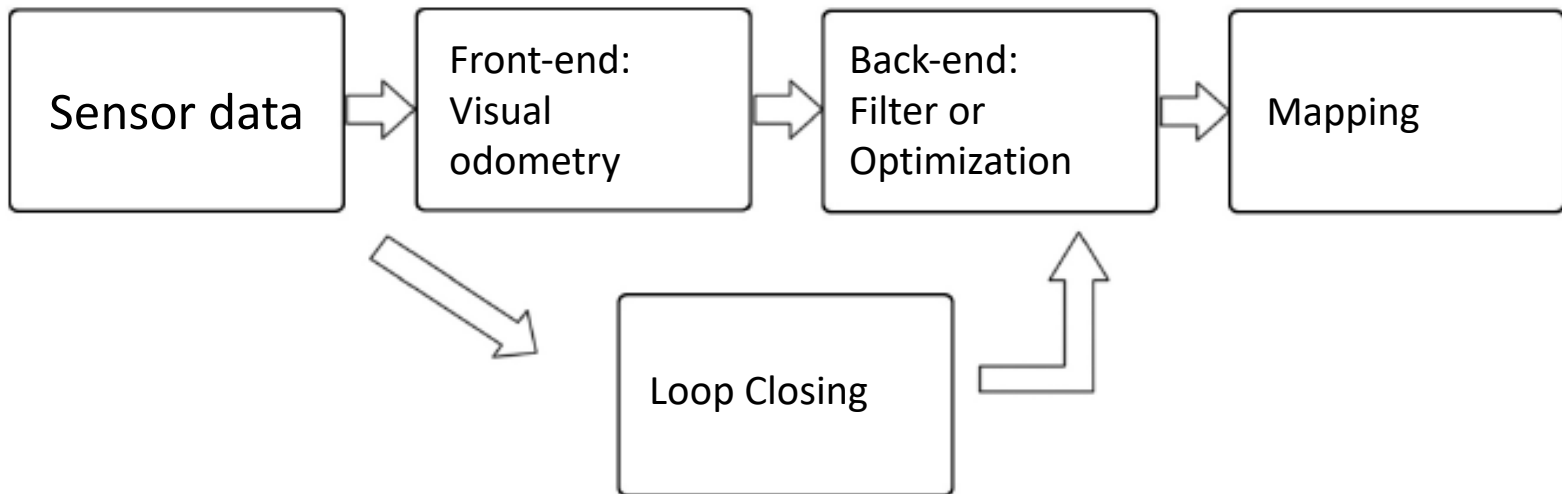
Moving stereo: disparity can be estimated in the motion



Ambiguity in mono vision: small + close or large + far away?

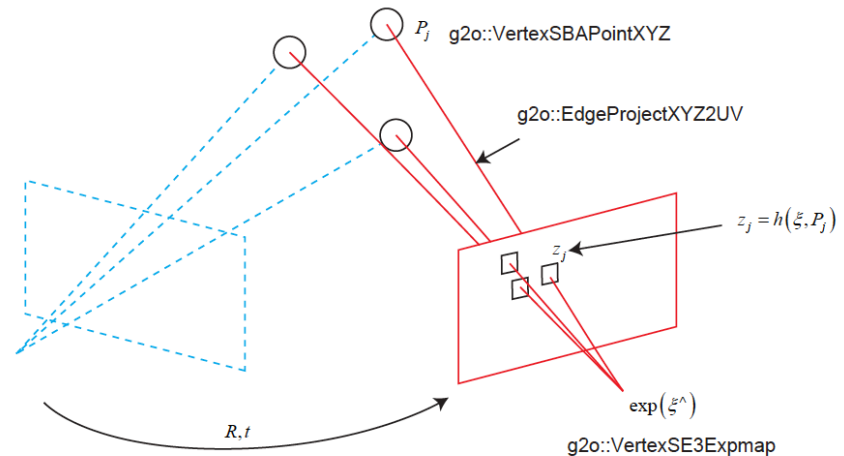
2. Framework of SLAM

- SLAM framework



2. Framework of SLAM

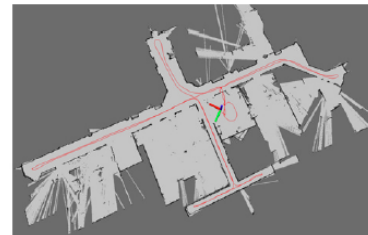
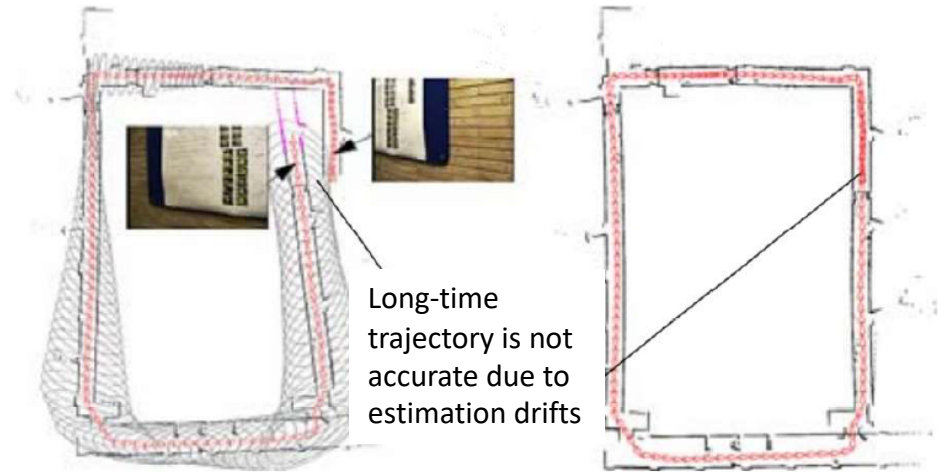
- Visual odometry
 - Motion estimation between adjacent frames
 - Simplest: two-view geometry
- Method
 - Feature method
 - Direct method
- Backend
 - Long-term trajectory and map estimation
 - MAP: Maximum of a Posteriori
 - Filters/Graph Optimization



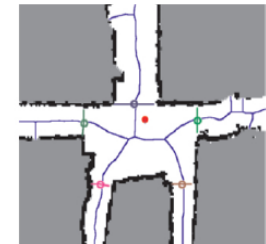
2. Framework of SLAM

- Loop closing
 - Correct the drift in estimation
 - Loop detection and correction

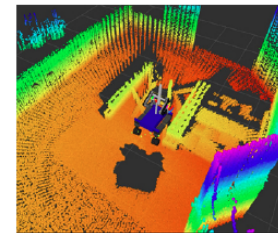
- Mapping
 - Generate globally consistent map for navigation/planning/communication/visualization etc
 - Grid/topological/hybrid maps
 - Pointcloud/Mesh/TSDF ...



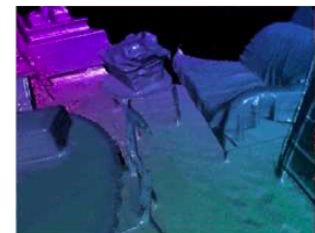
2D grid map



2D topological map



Point cloud maps



TSDF models

2. Framework of SLAM

- Mathematical representation of visual SLAM
- Assume a camera is moving in 3D space
 - But measurements are taken at discrete times:

$$\left\{ \begin{array}{l} \mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{w}_k) \\ \mathbf{z}_{k,j} = h(\mathbf{y}_j, \mathbf{x}_k, \mathbf{v}_{k,j}) \end{array} \right. \cdot \begin{array}{l} \text{Motion model} \\ \text{Observation model} \end{array}$$

Non-linear form

$$\left\{ \begin{array}{l} \mathbf{x}_k = A_k \mathbf{x}_{k-1} + B_k \mathbf{u}_k + \mathbf{w}_k \\ \mathbf{z}_{k,j} = C_j \mathbf{y}_j + D_k \mathbf{x}_k + \mathbf{v}_{k,j} \end{array} \right.$$

linear form

2. Framework of SLAM

- Questions:

$$\left\{ \begin{array}{l} \mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{w}_k) \\ z_{k,j} = h(\mathbf{y}_j, \mathbf{x}_k, \mathbf{v}_{k,j}) \end{array} \right. \cdot \begin{array}{l} \text{Motion model} \\ \text{Observation model} \end{array}$$

- How to represent state variables?
 - 3D geometry, Lie group and Lie algebra
- Exact form of motion/observation model?
 - Camera intrinsic and extrinsics
- How to estimate the state given measurement data?
 - State estimation problem
 - Filters and optimization

Contents

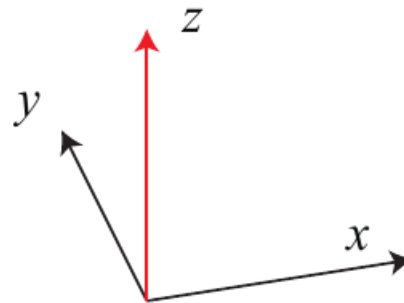
- Course contents and preliminary knowledge
- Framework and mathematic form of a SLAM problem
- 3D geometry
- Lie groups

3. 3D geometry

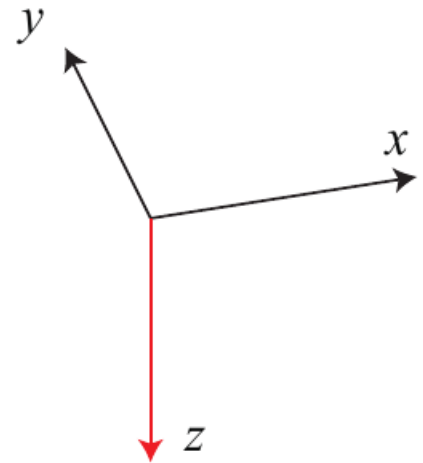
- Point and Coordinate system
- 2D: (x,y) and angle
- 3D?

3. 3D geometry

- 3D coordinate system
- Vectors and their coordinates



Right handed



Left handed

3. 3D geometry

- Vector operations
 - Addition/subtraction
 - Dot product

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{a}^T \mathbf{b} = \sum_{i=1}^3 a_i b_i = |\mathbf{a}| |\mathbf{b}| \cos \langle \mathbf{a}, \mathbf{b} \rangle .$$

- Cross product

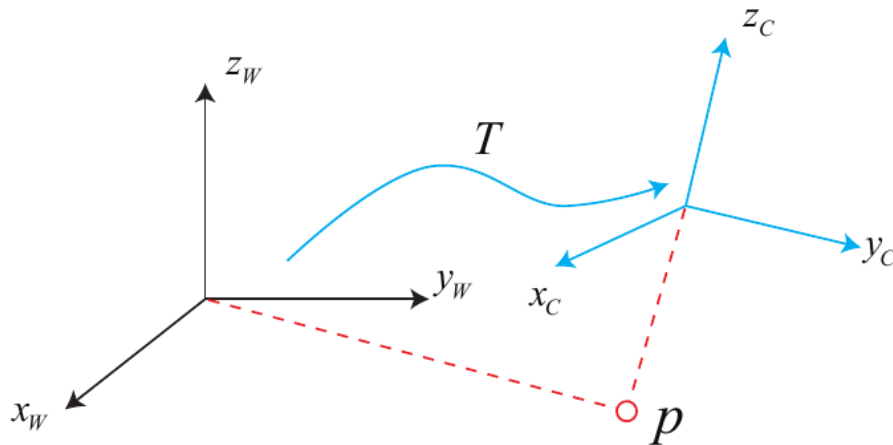
$$\mathbf{a} \times \mathbf{b} = \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{bmatrix} = \begin{bmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{bmatrix} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \mathbf{b} \triangleq \mathbf{a} \wedge \mathbf{b} .$$

Skew-symmetric operator

3. 3D geometry

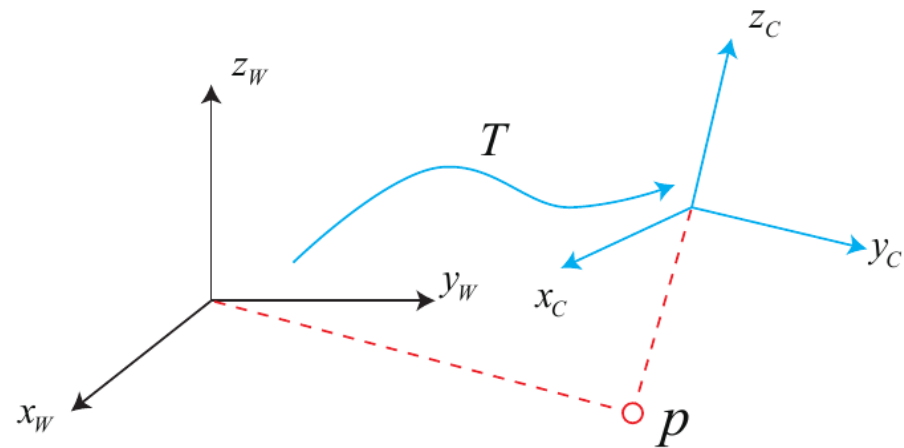
- Questions
 - Compute the coordinates in different systems?

- In SLAM:
 - Fixed world frame
 - Moving camera frame
 - Other sensor frames



3. 3D geometry

- 3D rigid body motion can be described with rotation and translation



- Translation is just a vector addition
- How to represent rotations?

3. 3D geometry

- Rotation

- Consider coordinate system (e_1, e_2, e_3) is rotated and become (e'_1, e'_2, e'_3)
- Vector a is fixed, then how are its coordinates changed?

$$[e_1, e_2, e_3] \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = [e'_1, e'_2, e'_3] \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \end{bmatrix} .$$

- Left multiplied by $[e_1^T, e_2^T, e_3^T]^T$

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} e_1^T e'_1 & e_1^T e'_2 & e_1^T e'_3 \\ e_2^T e'_1 & e_2^T e'_2 & e_2^T e'_3 \\ e_3^T e'_1 & e_3^T e'_2 & e_3^T e'_3 \end{bmatrix} \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \end{bmatrix} \triangleq Ra' .$$

Rotation matrix

3. 3D geometry

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} e_1^T e'_1 & e_1^T e'_2 & e_1^T e'_3 \\ e_2^T e'_1 & e_2^T e'_2 & e_2^T e'_3 \\ e_3^T e'_1 & e_3^T e'_2 & e_3^T e'_3 \end{bmatrix} \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \end{bmatrix} \triangleq \mathbf{R} a'.$$

- R is rotation matrix, which satisfies:
 - R is orthogonal
 - $\text{Det}(\mathbf{R}) = +1$ (if $\text{Det}(\mathbf{R}) = -1$ then it's improper rotation)
- Special orthogonal group:

$$SO(n) = \{ \mathbf{R} \in \mathbb{R}^{n \times n} \mid \mathbf{R}\mathbf{R}^T = \mathbf{I}, \det(\mathbf{R}) = 1 \}.$$

- Rotation from frame 2 to 1 can be written as:

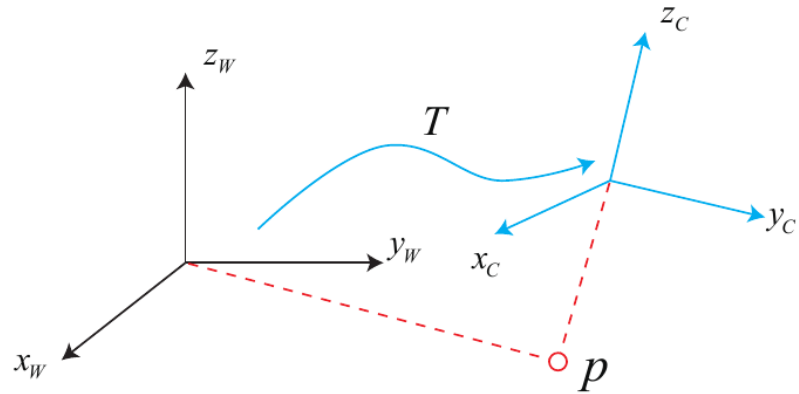
$$a_1 = R_{12} a_2 \quad \text{and vice versa:} \quad a_2 = R_{21} a_1$$

$$R_{21} = R_{12}^{-1} = R_{12}^T$$

3. 3D geometry

- Rotation plus translation:

$$\mathbf{a}' = \mathbf{R}\mathbf{a} + \mathbf{t}.$$



- Compounding rotation and translation:

- $b = \mathbf{R}_1\mathbf{a} + \mathbf{t}_1, \quad c = \mathbf{R}_2\mathbf{b} + \mathbf{t}_2. \quad \longrightarrow \quad c = \mathbf{R}_2(\mathbf{R}_1\mathbf{a} + \mathbf{t}_1) + \mathbf{t}_2.$

- Homogeneous form:

$$\begin{bmatrix} \mathbf{a}' \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ 1 \end{bmatrix} \triangleq \mathbf{T} \begin{bmatrix} \mathbf{a} \\ 1 \end{bmatrix}.$$

$$\tilde{\mathbf{b}} = \mathbf{T}_1\tilde{\mathbf{a}}, \quad \tilde{\mathbf{c}} = \mathbf{T}_2\tilde{\mathbf{b}} \quad \Rightarrow \quad \tilde{\mathbf{c}} = \mathbf{T}_2\mathbf{T}_1\tilde{\mathbf{a}}.$$

Inverse:
$$\mathbf{T}^{-1} = \begin{bmatrix} \mathbf{R}^T & -\mathbf{R}^T\mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}.$$

3. 3D geometry

- Homogenous coordinates:

$$\tilde{a} = \begin{bmatrix} a \\ 1 \end{bmatrix} \qquad \tilde{a} = \begin{bmatrix} a \\ 1 \end{bmatrix} = k \begin{bmatrix} a \\ 1 \end{bmatrix}$$

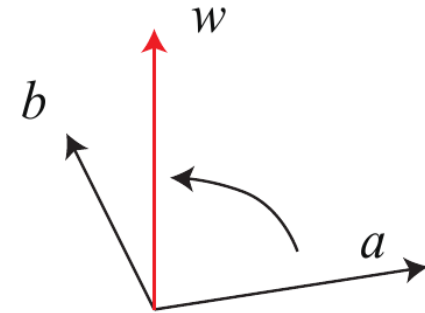
Still keeps equal when multiplying any non-zero factors

- Transform matrix forms Special Euclidean Group

$$SE(3) = \left\{ \mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \in \mathbb{R}^{4 \times 4} \mid \mathbf{R} \in SO(3), \mathbf{t} \in \mathbb{R}^3 \right\}.$$

3. 3D geometry

- Alternative rotation representations
 - Rotation vectors
 - Euler angles
 - Quaternions



- Rotation vectors
 - Angle + axis: θn
 - Rotation angle θ
 - Rotation axis n

Rotation vectors
Only three parameters

- Rotation vector to rotation matrix: **Rodrigues' formula**

$$\mathbf{R} = \cos \theta \mathbf{I} + (1 - \cos \theta) \mathbf{nn}^T + \sin \theta \mathbf{n}^\wedge.$$

- Inverse:

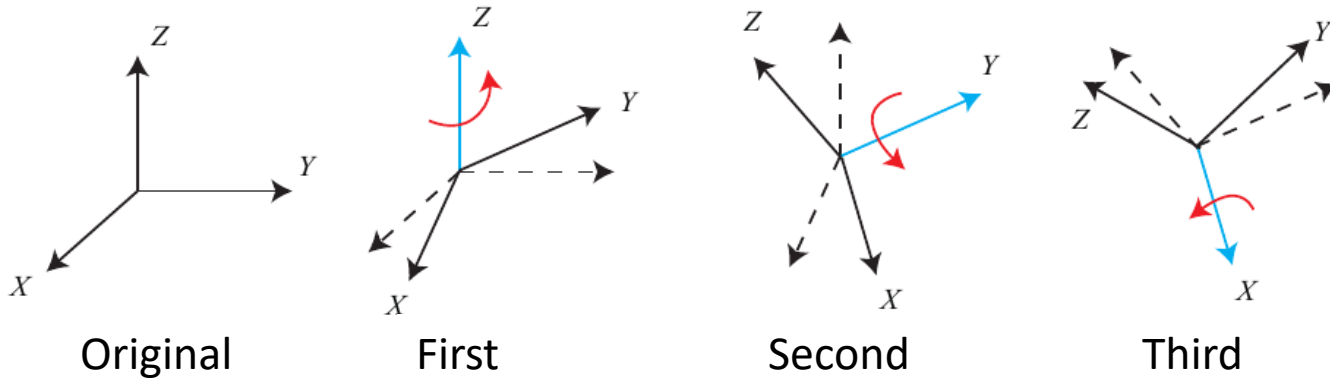
$$\theta = \arccos\left(\frac{\text{tr}(\mathbf{R}) - 1}{2}\right).$$

$$\mathbf{Rn} = n.$$

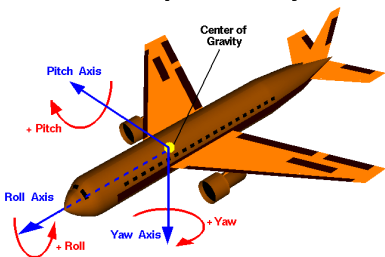
$$\log(\mathbf{R}) = \frac{\theta}{2 \sin \theta} (\mathbf{R} - \mathbf{R}^T)$$

3. 3D geometry

- Euler angles
 - Any rotation can be decomposed into three principal rotations



- However the order of axis can be defined very differently:
- Roll-pitch-yaw (in navigation) Spin-nutation-precession in mechanics



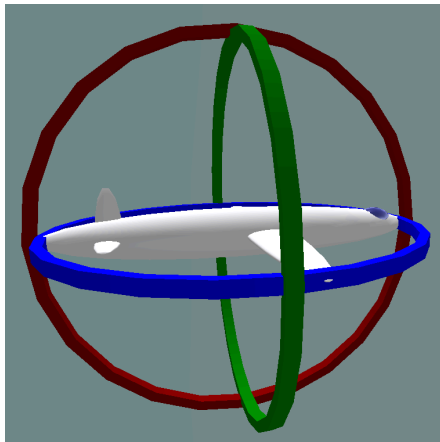
XYZ order

3-1-3 order

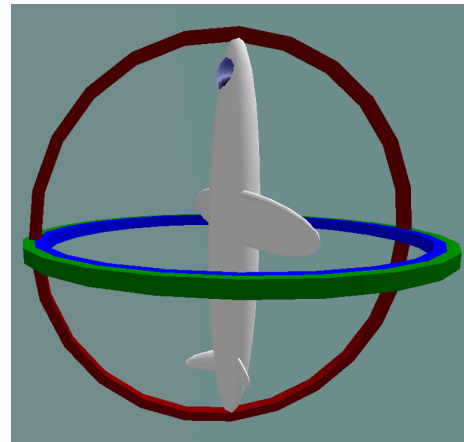
3. 3D geometry

- Gimbal lock
 - Singularity always exist if we want to use 3 parameters to describe rotation
 - Degree-of-Freedom is reduced in singular case
 - In yaw-pitch-roll order, when pitch=90 degrees

normal



singular



3. 3D geometry

- Quaternions

- In 2D case, we can use (unit) complex numbers to denote rotations

$$z = x + iy = \rho e^{i\theta}$$

Multiply i to rotate 90 degrees

- How about 3D case?

- (Unit) Quaternions

- Extended from complex numbers
- Three imaginary and one real part:
- The imaginary parts satisfy:

$$q = q_0 + q_1i + q_2j + q_3k,$$

$$\left\{ \begin{array}{l} i^2 = j^2 = k^2 = -1 \\ ij = k, ji = -k \\ jk = i, kj = -i \\ ki = j, ik = -j \end{array} \right.$$

i, j, k look like complex numbers when multiplying with themselves
And look like cross product when multiply with others

3. 3D geometry

- Quaternions

$$\mathbf{q} = q_0 + q_1i + q_2j + q_3k, \quad \mathbf{q} = [s, \mathbf{v}], \quad s = q_0 \in \mathbb{R}, \mathbf{v} = [q_1, q_2, q_3]^T \in \mathbb{R}^3,$$

- Operations

$$\mathbf{q}_a \pm \mathbf{q}_b = [s_a \pm s_b, \mathbf{v}_a \pm \mathbf{v}_b].$$

$$\mathbf{q}_a^* = s_a - x_a i - y_a j - z_a k = [s_a, -\mathbf{v}_a].$$

$$\begin{aligned} \mathbf{q}_a \mathbf{q}_b &= s_a s_b - x_a x_b - y_a y_b - z_a z_b \\ &\quad + (s_a x_b + x_a s_b + y_a z_b - z_a y_b) i \\ &\quad + (s_a y_b - x_a z_b + y_a s_b + z_a x_b) j \\ &\quad + (s_a z_b + x_a y_b - y_b x_a + z_a s_b) k. \end{aligned}$$

$$\|\mathbf{q}_a\| = \sqrt{s_a^2 + x_a^2 + y_a^2 + z_a^2}.$$

$$\mathbf{q}^{-1} = \mathbf{q}^* / \|\mathbf{q}\|^2.$$

$$k\mathbf{q} = [ks, k\mathbf{v}].$$

$$\mathbf{q}_a \mathbf{q}_b = [s_a s_b - \mathbf{v}_a^T \mathbf{v}_b, s_a \mathbf{v}_b + s_b \mathbf{v}_a + \mathbf{v}_a \times \mathbf{v}_b].$$

$$\mathbf{q}_a \cdot \mathbf{q}_b = s_a s_b + x_a x_b i + y_a y_b j + z_a z_b k.$$

3. 3D geometry

- From quaternions to angle-axis:

$$\mathbf{q} = \left[\cos \frac{\theta}{2}, n_x \sin \frac{\theta}{2}, n_y \sin \frac{\theta}{2}, n_z \sin \frac{\theta}{2} \right]^T .$$

- Inverse:

$$\begin{cases} \theta = 2 \arccos q_0 \\ [n_x, n_y, n_z]^T = [q_1, q_2, q_3]^T / \sin \frac{\theta}{2} \end{cases} .$$

- Rotate a vector by quaternions:

- Vector p is rotated by q and become p' , how to write their relationships?
- Write p as quaternion (pure imaginary): $\mathbf{p} = [0, x, y, z] = [0, \mathbf{v}]$.
- Then: $\mathbf{p}' = \mathbf{q}\mathbf{p}\mathbf{q}^{-1}$. Also pure imaginary

Contents

- Course contents and preliminary knowledge
- Framework and mathematic form of a SLAM problem
- 3D geometry
- Lie groups

4. Lie Group and Lie Algebra

- Recall the mathematic model of SLAM

$$\begin{cases} \mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{w}_k) & \text{Motion model} \\ \mathbf{z}_{k,j} = h(\mathbf{y}_j, \mathbf{x}_k, \mathbf{v}_{k,j}) & \text{Observation model} \end{cases} .$$

- We use $SO(3)$ and $SE(3)$ to represent the pose of camera
- Let's consider optimizing some function of rotation/transform

$$f(R) \quad \frac{df}{dR} \quad \frac{f(R + \Delta R) - f(R)}{\Delta R}$$

- Rotation and transform matrix don't have a plus operator!

4. Lie Group and Lie Algebra

- Group
 - 3D rotation matrix forms the Special Orthogonal Group

$$SO(3) = \{R \in \mathbb{R}^{3 \times 3} | RR^T = I, \det(R) = 1\}.$$

- 3D transform matrix forms the Special Euclidean Group

$$SE(3) = \left\{ T = \begin{bmatrix} R & t \\ \mathbf{0}^T & 1 \end{bmatrix} \in \mathbb{R}^{4 \times 4} | R \in SO(3), t \in \mathbb{R}^3 \right\}.$$

- What is Group?

4. Lie Group and Lie Algebra

- Group

- Group is a set with an operator (A, \cdot) that satisfies the following:

1. Closure $\forall a_1, a_2 \in A, \quad a_1 \cdot a_2 \in A.$

2. Associativity $\forall a_1, a_2, a_3 \in A, \quad (a_1 \cdot a_2) \cdot a_3 = a_1 \cdot (a_2 \cdot a_3).$

3. Identity $\exists a_0 \in A, \quad s.t. \quad \forall a \in A, \quad a_0 \cdot a = a \cdot a_0 = a.$

4. Invertibility $\forall a \in A, \quad \exists a^{-1} \in A, \quad s.t. \quad a \cdot a^{-1} = a_0.$

- Obviously,

- $(SO(3), \cdot), (SE(3), \cdot)$ are groups

4. Lie Group and Lie Algebra

- Lie Group
 - Group that is smooth
 - Group that is also a manifold
 - “Locally looks like R^n ”
 - Further explanation needs knowledge from topology and differential geometry
 - $SO(3)$ and $SE(3)$ are also Lie groups
- Lie Algebra
 - Tangent space of the Lie group at identity
 - $SO(3) \rightarrow so(3)$, $SE(3) \rightarrow se(3)$

4. Lie Group and Lie Algebra

- Introducing of the Lie Algebra
 - Assume a time-varying rotation matrix $R(t)$
 - It satisfies: $R(t)R(t)^T = I$.

- Take derivative of time t at both sides:

$$\dot{R}(t)R(t)^T + R(t)\dot{R}(t)^T = 0.$$

- Rearrange: $\dot{R}(t)R(t)^T = -\left(\dot{R}(t)R(t)^T\right)^T$.

Skew-symmetric

4. Lie Group and Lie Algebra

$$\mathbf{a}^\wedge = \mathbf{A} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}, \quad \mathbf{A}^\vee = \mathbf{a}.$$

- Denote the skew-symmetric matrix as $\phi(t)^\wedge$

$$\dot{\mathbf{R}}(t)\mathbf{R}(t)^T = -\left(\dot{\mathbf{R}}(t)\mathbf{R}(t)^T\right)^T. \quad \dot{\mathbf{R}}(t)\mathbf{R}(t)^T = \phi(t)^\wedge.$$

- Put $\mathbf{R}(t)$ to the right side: $\dot{\mathbf{R}}(t) = \phi(t)^\wedge \mathbf{R}(t)$
- It looks like when we take the derivative, we will get a $\phi(t)^\wedge$ at the left side
- Assume we are close to identity: $t_0 = 0, \mathbf{R}(0) = \mathbf{I}$
- And $\phi(t)^\wedge$ does not change: $\dot{\mathbf{R}}(t) = \phi(t_0)^\wedge \mathbf{R}(t) = \phi_0^\wedge \mathbf{R}(t)$.
- With $\mathbf{R}(0) = \mathbf{I}$, we solve this ODE:

$$\mathbf{R}(t) = \exp(\phi_0^\wedge t).$$

4. Lie Group and Lie Algebra



- So, if t is close to 0, then we can always find an R given ϕ
- ϕ is called a Lie algebra
- From a Lie algebra, if we take a **Exponential Map**, then it becomes a Lie group
- Questions:
 - Lie algebra's definition and constraints?
 - How to compute the exponential map?

4. Lie Group and Lie Algebra

- Lie algebra:
 - We have a Lie algebra for each Lie group, which is a vector space (the tangent space) at the identity
 - Lie algebra has a vector space V over field F together with a binary operator (Lie bracket) $[\cdot, \cdot]$, that satisfies:
 - Closure: $\forall X, Y \in \mathbb{V}. [X, Y] \in \mathbb{V}.$
 - Bilinearity: for any $\forall X, Y, Z \in \mathbb{V}, a, b \in \mathbb{F},$

$$[aX + bY, Z] = a[X, Z] + b[Y, Z], \quad [Z, aX + bY] = a[Z, X] + b[Z, Y].$$

- Alternativity: $\forall X \in \mathbb{V}, [X, X] = \mathbf{0}.$
- Jacobi identity:

$$\forall X, Y, Z \in \mathbb{V}, [X, [Y, Z]] + [Z, [X, Y]] + [Y, [Z, X]] = \mathbf{0}.$$

4. Lie Group and Lie Algebra

- Example: (R^3, R, \times) is a Lie algebra
- Lie algebra $\mathfrak{so}(3)$: $\mathfrak{so}(3) = \{\phi \in \mathbb{R}^3, \Phi = \phi^\wedge \in \mathbb{R}^{3 \times 3}\}$.

- where

$$\Phi = \phi^\wedge = \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix} \in \mathbb{R}^{3 \times 3}.$$

- And the Lie bracket is: $[\phi_1, \phi_2] = (\Phi_1 \Phi_2 - \Phi_2 \Phi_1)^\vee$.

4. Lie Group and Lie Algebra

- Similarly, for SE(3) we also have se(3):

$$\mathfrak{se}(3) = \left\{ \xi = \begin{bmatrix} \rho \\ \phi \end{bmatrix} \in \mathbb{R}^6, \rho \in \mathbb{R}^3, \phi \in \mathfrak{so}(3), \xi^\wedge = \begin{bmatrix} \phi^\wedge & \rho \\ \mathbf{0}^T & 0 \end{bmatrix} \in \mathbb{R}^{4 \times 4} \right\}.$$

- Where

$$\xi^\wedge = \begin{bmatrix} \phi^\wedge & \rho \\ \mathbf{0}^T & 0 \end{bmatrix} \in \mathbb{R}^{4 \times 4}.$$

and Lie bracket is:

$$[\xi_1, \xi_2] = (\xi_1^\wedge \xi_2^\wedge - \xi_2^\wedge \xi_1^\wedge)^\vee.$$

NOTE in se(3) this operator is not a skew-symmetric matrix, but we still keeps its form

- Note:
 - The definition of se(3) may be different in literature
 - Vector or matrix are both ok to define a lie algebra

4. Lie Group and Lie Algebra

- Exponential map
 - Operator from Lie algebra to Lie group: $R = \exp(\phi^\wedge)$
 - Here ϕ^\wedge is a 3x3 matrix so this exponential map is a matrix operator
 - Take Taylor expansion:

$$\exp(\phi^\wedge) = \sum_{n=0}^{\infty} \frac{1}{n!} (\phi^\wedge)^n.$$

- Directly computing this Taylor expansion is intractable

4. Lie Group and Lie Algebra

- Take the length and direction of ϕ , then $\phi = \theta a$
- For a unit-length vector, we have:

$$a^{\wedge} a^{\wedge} = a a^T - I,$$

$$a^{\wedge} a^{\wedge} a^{\wedge} = -a^{\wedge}.$$

This will be useful when handling the high-order Taylor expansion items

4. Lie Group and Lie Algebra

- Compute the Taylor expansion:

$$\begin{aligned}\exp(\phi^\wedge) &= \exp(\theta \mathbf{a}^\wedge) = \sum_{n=0}^{\infty} \frac{1}{n!} (\theta \mathbf{a}^\wedge)^n \\ &= \mathbf{I} + \theta \mathbf{a}^\wedge + \frac{1}{2!} \theta^2 \mathbf{a}^\wedge \mathbf{a}^\wedge + \frac{1}{3!} \theta^3 \mathbf{a}^\wedge \mathbf{a}^\wedge \mathbf{a}^\wedge + \frac{1}{4!} \theta^4 (\mathbf{a}^\wedge)^4 + \dots \\ &= \mathbf{a} \mathbf{a}^T - \mathbf{a}^\wedge \mathbf{a}^\wedge + \theta \mathbf{a}^\wedge + \frac{1}{2!} \theta^2 \mathbf{a}^\wedge \mathbf{a}^\wedge - \frac{1}{3!} \theta^3 \mathbf{a}^\wedge - \frac{1}{4!} \theta^4 (\mathbf{a}^\wedge)^2 + \dots \\ &= \mathbf{a} \mathbf{a}^T + \left(\theta - \frac{1}{3!} \theta^3 + \frac{1}{5!} \theta^5 - \dots \right) \mathbf{a}^\wedge - \left(1 - \frac{1}{2!} \theta^2 + \frac{1}{4!} \theta^4 - \dots \right) \mathbf{a}^\wedge \mathbf{a}^\wedge \\ &= \mathbf{a}^\wedge \mathbf{a}^\wedge + \mathbf{I} + \sin \theta \mathbf{a}^\wedge - \cos \theta \mathbf{a}^\wedge \mathbf{a}^\wedge \\ &= (1 - \cos \theta) \mathbf{a}^\wedge \mathbf{a}^\wedge + \mathbf{I} + \sin \theta \mathbf{a}^\wedge \\ &= \cos \theta \mathbf{I} + (1 - \cos \theta) \mathbf{a} \mathbf{a}^T + \sin \theta \mathbf{a}^\wedge.\end{aligned}$$

- Finally we get:

$$\exp(\theta \mathbf{a}^\wedge) = \cos \theta \mathbf{I} + (1 - \cos \theta) \mathbf{a} \mathbf{a}^T + \sin \theta \mathbf{a}^\wedge.$$

- Which is exactly the Rodrigues' formula!

4. Lie Group and Lie Algebra

- So $\mathfrak{so}(3)$ is just the rotation vector
- Same as exponential map, we can also define logarithm map as:

$$\phi = \ln(\mathbf{R})^\vee = \left(\sum_{n=0}^{\infty} \frac{(-1)^n}{n+1} (\mathbf{R} - \mathbf{I})^{n+1} \right)^\vee.$$

- And also don't need to actually compute this stuff, we take the conversion equations from rotation matrix to rotation vector:

$$\theta = \arccos\left(\frac{\text{tr}(\mathbf{R}) - 1}{2}\right).$$

$$Rn = n.$$

$$\log(\mathbf{R}) = \frac{\theta}{2 \sin \theta} (\mathbf{R} - \mathbf{R}^\top)$$

4. Lie Group and Lie Algebra

- For SE(3), the exponential map is:

$$\begin{aligned} \exp(\xi^\wedge) &= \begin{bmatrix} \sum_{n=0}^{\infty} \frac{1}{n!} (\phi^\wedge)^n & \sum_{n=0}^{\infty} \frac{1}{(n+1)!} (\phi^\wedge)^n \rho \\ \mathbf{0}^T & 1 \end{bmatrix} \\ &\triangleq \begin{bmatrix} R & J\rho \\ \mathbf{0}^T & 1 \end{bmatrix} = T. \end{aligned}$$

- The rotation part is just a SO(3), but the translation part has a Jacobian matrix: (left as an assignment)

$$J = \frac{\sin \theta}{\theta} I + \left(1 - \frac{\sin \theta}{\theta}\right) \mathbf{a} \mathbf{a}^T + \frac{1 - \cos \theta}{\theta} \mathbf{a}^\wedge.$$

4. Lie Group and Lie Algebra

Rotation matrix

Lie group

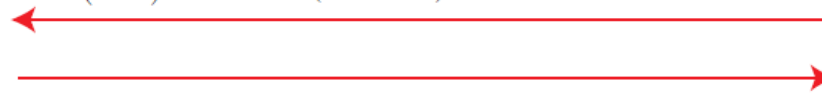
$$SO(3)$$

$$R \in \mathbb{R}^{3 \times 3}$$

$$RR^T = I$$

$$\det(R) = 1$$

$$\exp(\theta a^\wedge) = \cos \theta I + (1 - \cos \theta) a a^T + \sin \theta a^\wedge \quad \text{Exponential}$$



$$\text{Logarithm} \quad \theta = \arccos \frac{\text{tr}(R) - 1}{2} \quad Ra = a$$

Lie algebra

$$\mathfrak{so}(3)$$

$$\phi \in \mathbb{R}^3$$

$$\phi^\wedge = \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix}$$

Transform matrix

Lie group

$$SE(3)$$

$$T \in \mathbb{R}^{4 \times 4}$$

$$T = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix}$$

$$\exp(\xi^\wedge) = \begin{bmatrix} \exp(\phi^\wedge) & J\rho \\ 0^T & 1 \end{bmatrix}$$

$$J = \frac{\sin \theta}{\theta} I + \left(1 - \frac{\sin \theta}{\theta}\right) a a^T + \frac{1 - \cos \theta}{\theta} a^\wedge \quad \text{Exponential}$$



$$\text{Logarithm} \quad \theta = \arccos \frac{\text{tr}(R) - 1}{2} \quad Ra = a \quad t = J\rho$$

Lie algebra

$$\mathfrak{se}(3)$$

$$\xi = \begin{bmatrix} \rho \\ \phi \end{bmatrix} \in \mathbb{R}^6$$

$$\xi^\wedge = \begin{bmatrix} \phi^\wedge & \rho \\ 0^T & 0 \end{bmatrix}$$

4. Lie Group and Lie Algebra

- Next question
 - We still don't have plus operation for Lie group
 - Then we can't define derivatives
- Solution
 - Take advantage of the plus in the Lie algebra, and convert it back to Lie group
- A primal question:
 - Plus in Lie algebra is equal to multiplication in Lie group?

$$\exp(\phi_1^\wedge) \exp(\phi_2^\wedge) = \exp((\phi_1 + \phi_2)^\wedge). \quad ?$$

4. Lie Group and Lie Algebra

- Unfortunately, this does not work for matrices
- Baker-Campbell-Hausdorff formula gives the full version of this multiplication:

$$\begin{aligned} & \ln(\exp(\mathbf{A}) \exp(\mathbf{B})) \\ &= \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n} \sum_{\substack{r_i + s_i > 0, \\ 1 \leq i \leq n}} \frac{(\sum_{i=1}^n (r_i + s_i))^{-1}}{\prod_{i=1}^n r_i! s_i!} [\mathbf{A}^{r_1} \mathbf{B}^{s_1} \mathbf{A}^{r_2} \mathbf{B}^{s_2} \dots \mathbf{A}^{r_n} \mathbf{B}^{s_n}] \end{aligned}$$

- where

$$\begin{aligned} & [\mathbf{A}^{r_1} \mathbf{B}^{s_1} \mathbf{A}^{r_2} \mathbf{B}^{s_2} \dots \mathbf{A}^{r_n} \mathbf{B}^{s_n}] \\ &= \underbrace{[\mathbf{A}, \dots [\mathbf{A}, \mathbf{B}, \dots [\mathbf{B}, \dots [\mathbf{A}, \dots [\mathbf{A}, \mathbf{B}, \dots [\mathbf{B}, \mathbf{B}] \dots]] \dots] \dots] \dots]}_{r_1} \underbrace{\dots}_{s_1} \underbrace{\dots}_{r_n} \underbrace{\dots}_{s_n} \end{aligned}$$

4. Lie Group and Lie Algebra

- First part of BCH formula:

$$\ln(\exp(A)\exp(B)) = A + B + \frac{1}{2}[A, B] + \frac{1}{12}[A, [A, B]] - \frac{1}{12}[B, [A, B]] + \dots$$

- If A or B is small enough we can keep the linear item only, the BCH can be approximately written as:

$$\ln(\exp(\phi_1^\wedge)\exp(\phi_2^\wedge))^\vee \approx \begin{cases} J_l(\phi_2)^{-1}\phi_1 + \phi_2 & \text{if } \phi_1 \text{ is small,} \\ J_r(\phi_1)^{-1}\phi_2 + \phi_1 & \text{if } \phi_2 \text{ is small.} \end{cases}$$

- where $J_l = J = \frac{\sin \theta}{\theta} \mathbf{I} + \left(1 - \frac{\sin \theta}{\theta}\right) \mathbf{a}\mathbf{a}^T + \frac{1 - \cos \theta}{\theta} \mathbf{a}^\wedge$. Left Jacobian
- $J_l^{-1} = \frac{\theta}{2} \cot \frac{\theta}{2} \mathbf{I} + \left(1 - \frac{\theta}{2} \cot \frac{\theta}{2}\right) \mathbf{a}\mathbf{a}^T - \frac{\theta}{2} \mathbf{a}^\wedge$. Right Jacobian

$$J_r(\phi) = J_l(-\phi).$$

4. Lie Group and Lie Algebra

- Rewrite it (we take left multiplication as an example)

$$\exp(\Delta\phi^\wedge) \exp(\phi^\wedge) = \exp\left(\left(\phi + J_l^{-1}(\phi) \Delta\phi\right)^\wedge\right).$$

- Left multiplication in Lie group means an addition in Lie algebra with an Jacobian
- Inversely, if we do addition in Lie algebra, the in Lie group:

$$\exp\left(\left(\phi + \Delta\phi\right)^\wedge\right) = \exp\left(\left(J_l \Delta\phi\right)^\wedge\right) \exp(\phi^\wedge) = \exp(\phi^\wedge) \exp\left(\left(J_r \Delta\phi\right)^\wedge\right).$$

4. Lie Group and Lie Algebra

- Similar in SE(3)'s case:

$$\exp(\Delta \hat{\xi}) \exp(\hat{\xi}) \approx \exp\left(\left(\mathcal{J}_l^{-1} \Delta \xi + \xi\right)^\wedge\right),$$

$$\exp(\hat{\xi}) \exp(\Delta \hat{\xi}) \approx \exp\left(\left(\mathcal{J}_r^{-1} \Delta \xi + \xi\right)^\wedge\right).$$

- Where:

$$\mathcal{J}_r(\xi) = \begin{bmatrix} J_r & Q_r \\ \mathbf{0} & J_r \end{bmatrix}$$

$$\mathcal{J}_l(\xi) = \begin{bmatrix} J_l & Q_l \\ \mathbf{0} & J_l \end{bmatrix}$$

$$\begin{aligned} Q_l(\xi) &= \sum_{n=0}^{\infty} \sum_{m=0}^{\infty} \frac{1}{(n+m+2)!} (\phi^\wedge)^n \rho^\wedge (\phi^\wedge)^m \\ &= \frac{1}{2} \rho^\wedge + \left(\frac{\phi - \sin \phi}{\phi^3} \right) (\phi^\wedge \rho^\wedge + \rho^\wedge \phi^\wedge + \phi^\wedge \rho^\wedge \phi^\wedge) \\ &\quad + \left(\frac{\phi^2 + 2 \cos \phi - 2}{2\phi^4} \right) (\phi^\wedge \phi^\wedge \rho^\wedge + \rho^\wedge \phi^\wedge \phi^\wedge - 3\phi^\wedge \rho^\wedge \phi^\wedge) \\ &\quad + \left(\frac{2\phi - 3 \sin \phi + \phi \cos \phi}{2\phi^5} \right) (\phi^\wedge \rho^\wedge \phi^\wedge \phi^\wedge + \phi^\wedge \phi^\wedge \rho^\wedge \phi^\wedge) \end{aligned}$$

$$Q_r(\xi) = Q_l(-\xi) = C Q_l(\xi) + (J_l \rho)^\wedge C J_l$$

4. Lie Group and Lie Algebra

- With BCH formula, we can define the derivative of a function of a rotation or transform matrix
- Example: rotating a point p
- We want to know the derivative: $\frac{\partial (Rp)}{\partial R}$.
- We have two solutions:
 - Add a small item in the Lie algebra, and set its limit to zero (Derivative model)
 - (Left) Multiply a small item in the Lie group, and set its Lie algebra's limit to zero (Disturb model)

4. Lie Group and Lie Algebra

- Derivative model:

$$\begin{aligned}\frac{\partial (\exp(\phi^\wedge) p)}{\partial \phi} &= \lim_{\delta \phi \rightarrow 0} \frac{\exp((\phi + \delta \phi)^\wedge) p - \exp(\phi^\wedge) p}{\delta \phi} \\ &= \lim_{\delta \phi \rightarrow 0} \frac{\exp((J_l \delta \phi)^\wedge) \exp(\phi^\wedge) p - \exp(\phi^\wedge) p}{\delta \phi} \\ &\approx \lim_{\delta \phi \rightarrow 0} \frac{(I + (J_l \delta \phi)^\wedge) \exp(\phi^\wedge) p - \exp(\phi^\wedge) p}{\delta \phi} \\ &= \lim_{\delta \phi \rightarrow 0} \frac{(J_l \delta \phi)^\wedge \exp(\phi^\wedge) p}{\delta \phi} \\ &= \lim_{\delta \phi \rightarrow 0} \frac{-(\exp(\phi^\wedge) p)^\wedge J_l \delta \phi}{\delta \phi} = -(Rp)^\wedge J_l.\end{aligned}$$

4. Lie Group and Lie Algebra

- Disturb model:

$$\begin{aligned}\frac{\partial (Rp)}{\partial \varphi} &= \lim_{\varphi \rightarrow 0} \frac{\exp(\varphi^\wedge) \exp(\phi^\wedge) p - \exp(\phi^\wedge) p}{\varphi} \\ &\approx \lim_{\varphi \rightarrow 0} \frac{(1 + \varphi^\wedge) \exp(\phi^\wedge) p - \exp(\phi^\wedge) p}{\varphi} \\ &= \lim_{\varphi \rightarrow 0} \frac{\varphi^\wedge Rp}{\varphi} = \lim_{\varphi \rightarrow 0} \frac{-(Rp)^\wedge \varphi}{\varphi} = -(Rp)^\wedge.\end{aligned}$$

- More simple and clear
- In some literature we use operator \oplus to denote this disturb model

$$\Delta R \oplus R = \exp(\delta\phi^\wedge)R$$

4. Lie Group and Lie Algebra

- Disturb model in SE(3):

$$\begin{aligned}
 \frac{\partial (Tp)}{\partial \delta \xi} &= \lim_{\delta \xi \rightarrow 0} \frac{\exp(\delta \xi^\wedge) \exp(\xi^\wedge) p - \exp(\xi^\wedge) p}{\delta \xi} \\
 &\approx \lim_{\delta \xi \rightarrow 0} \frac{(I + \delta \xi^\wedge) \exp(\xi^\wedge) p - \exp(\xi^\wedge) p}{\delta \xi} \\
 &= \lim_{\delta \xi \rightarrow 0} \frac{\delta \xi^\wedge \exp(\xi^\wedge) p}{\delta \xi} \\
 &= \lim_{\delta \xi \rightarrow 0} \frac{\begin{bmatrix} \delta \phi^\wedge & \delta \rho \\ \mathbf{0}^T & 0 \end{bmatrix} \begin{bmatrix} Rp + t \\ 1 \end{bmatrix}}{\delta \xi} \\
 &= \lim_{\delta \xi \rightarrow 0} \frac{\begin{bmatrix} \delta \phi^\wedge (Rp + t) + \delta \rho \\ 0 \end{bmatrix}}{\delta \xi} = \begin{bmatrix} I & -(Rp + t)^\wedge \\ \mathbf{0}^T & \mathbf{0}^T \end{bmatrix} \triangleq (Tp)^\odot.
 \end{aligned}$$

Questions?