Name: Cuong Ha; Matriculation Number: 03749410

Part 1: CMake

• **set**(CMAKE_MODULE_PATH "${CMAKE_CURRENT_SOURCE_DIR}/cmake_modules/" ${
CMAKE_MODULE_PATH})
    // specify a search path for CMake modules to be loaded by the the include() or find_package()

• **set**(CMAKE_CXX_STANDARD 14)
  set(CMAKE_CXX_STANDARD_REQUIRED ON)
  set(CMAKE_CXX_EXTENSIONS OFF)
    // Specify that C++ 14 standard is required, disable C++ extensions (e.g. use e.g. -std=c++11 rather than -std=gnu++11) for broadest compatibility across compilers

• **set**(CMAKE_CXX_FLAGS_DEBUG "-O0 -g -DEIGEN_INITIALIZE_MATRICES_BY_NAN")
    // set compile flag for building in Debug mode: optimize level 0 and init Eigen matrix by Nan to track for the problem of uninitialized data being used.

  set(CMAKE_CXX_FLAGS_RELWITHDEBINFO "-O3 -DNDEBUG -g -
DEIGEN_INITIALIZE_MATRICES_BY_NAN")
    // set compile flag for building in Release with Debug information mode: Optimization level 3, disable assert, and debug info a.k.a additional symbol table information for debugging and tracking for uninitialized Eigen matrix.

  set(CMAKE_CXX_FLAGS_RELEASE "-O3 -DNDEBUG")
    // set compile flag for building in Release mode: Optimization level 3, disable assert.

  SET(CMAKE_CXX_FLAGS " -ftemplate -backtrace -limit=0 -Wall -Wextra ${
EXTRA_WARNING_FLAGS} -march=${CXX_MARCH} ${CMAKE_CXX_FLAGS}")
    // Default compile flag

• **add_executable**(calibration src/calibration.cpp)
  target_link_libraries(calibration Ceres::ceres pangolin TBB)
    // Create executable file “calibration” from source file and link it with libraries: Ceres, pangolin and TBB.

Part 2: Prove SE3 Exponential map

Exponential map for SE(3).

- Given a continous family of rigid-body transformation

$$g : \mathbb{R} \to SE(3) ; \quad g(t) = \begin{pmatrix} R(t) & T(t) \\ 0 & 1 \end{pmatrix} \in \mathbb{R}^{4\times 4}$$

- We have inverse transform :

$$g^{-1}(t) = \begin{pmatrix} R^{-1} & -R^{-1}T \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} R^T & -R^T T \\ 0 & 1 \end{pmatrix}$$

- We consider:

$$\dot{g}(t)\, g^{-1}(t) = \begin{pmatrix} \dot{R} & \dot{T} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} R^T & -R^T T \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \dot{R}R^T & \dot{T} - \dot{R}R^T T \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{4\times 4}$$

- As in the case of SO(3), $\dot{R}R^T$ corresponds to some skew symmetric matric $\hat{w} \in so(3)$

Defining a vector $v(t) = \dot{T} - \hat{w}(t)T(t)$ we have:

$$\dot{g}(t) \cdot g^{-1}(t) = \begin{pmatrix} \hat{w}(t) & v(t) \\ 0 & 0 \end{pmatrix} \equiv \hat{\xi}(t) \in \mathbb{R}^{4\times 4}$$

- Furthermore

$$\dot{g}(t) \cdot g^{-1}(t) \cdot g(t) = \dot{g}(t) = \hat{\xi}(t) \cdot g(t)$$

Hence we have differential equation system:

$$\begin{cases} \dot{g}(t) = \hat{\xi}\, g(t) , \quad \hat{\xi} = const \\ g(0) = I \end{cases}$$

It has the solution : $g(t) = e^{\hat{\xi} t} = \sum_{n=0}^{\infty} \frac{(\hat{\xi} t)^n}{n!}$

where $\hat{\xi} = \begin{pmatrix} \hat{w} & v \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{4\times 4}$ and $\xi = \begin{pmatrix} v \\ w \end{pmatrix} \in \mathbb{R}^6$ (1)

We have $\hat{\xi}^2 = \begin{pmatrix} \hat{w} & v \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \hat{w} & v \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} \hat{w}^2 & \hat{w} v \\ 0 & 0 \end{pmatrix}$

By induction, $\hat{\xi}^n = \hat{\xi}^{n-1} . \hat{\xi} = \begin{pmatrix} \hat{w}^{n-1} & \hat{w}^{n-2} . v \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \hat{w} & v \\ 0 & 0 \end{pmatrix}$

$$= \begin{pmatrix} \hat{w}^n & \hat{w}^{n-1} . v \\ 0 & 0 \end{pmatrix}$$

Then : $g(t) = e^{\hat{\xi} t} = \begin{pmatrix} \sum_{n=0}^{\infty} \hat{w}^n \frac{1}{n!} & \sum_{n=1}^{\infty} \frac{1}{n!} \hat{w}^{n-1} . v \\ 0 & 1 \end{pmatrix}$

$$= \begin{pmatrix} \sum_{n=0}^{\infty} \hat{w}^n . \frac{1}{n!} & \left(\sum_{n=0}^{\infty} \frac{1}{(n+1)!} \hat{w}^n \right) . v \\ 0 & 1 \end{pmatrix} \quad (2)$$

In addition, let $\theta = |w|$ and $u = w / |w|$: $\hat{u}^2 = u u^T - I$; $\hat{u}^3 = -\hat{u}$

$$\sum_{n=0}^{\infty} \frac{1}{(n+1)!} (\hat{w})^n = \sum_{n=0}^{\infty} \frac{1}{(n+1)!} (\hat{u} \theta)^n$$

$$= I + \frac{1}{\theta} \left(\frac{\theta^2}{2!} - \frac{\theta^4}{4!} + \frac{\theta^6}{6!} - \dots \right) \hat{u}$$

$$+ \frac{1}{\theta} \left(\frac{\theta^3}{3!} - \frac{\theta^5}{5!} + \frac{\theta^7}{7!} - \dots \right) . \hat{u}^2$$

$$= I + \frac{1}{\theta} . (1 - \cos \theta) . \hat{u} + \frac{1}{\theta} (\theta - \sin \theta) . \hat{u}^2$$

$$= I + \frac{1 - \cos(\theta)}{\theta^2} . \hat{w} + \frac{\theta - \sin \theta}{\theta^3} . \hat{w}^2 = J \quad (3)$$

From (1) and (2) and (3) we have proof foro exponential map of SE3.

Part 3: What is SLAM?
From Cesar Cadena et al. “Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age”.

1. Why would a SLAM system need a map?
A SLAM system requires the use of a map for two reasons:
- To support other tasks such as informing path planning or providing intuitive visualization for a human operator.
- Limit the error committed in estimating the state of the robot. Without a map, dead-reckoning would quickly drift over time; on the other hand, using a map, e.g., a set of distinguishable landmarks, the robot can “reset” its localization error by re-visiting known areas (so-called loop closure).

2. How can we apply SLAM technology to real-world applications?
- SLAM finds applications in all scenarios in which a prior map is not available and needs to be built.
- It raised in popularity in indoor applications of mobile robotics, where GPS is inaccurate.
- Furthermore, SLAM provides an appealing alternative to user-built maps, showing that robot operation is possible in the absence of an ad hoc localization infrastructure.

3. Describe the history of SLAM.
History of SLAM can be splitted into two ages:
- Classical age: 1986-2004
    - introduction of the main probabilistic formulations for SLAM, including approaches based on Extended Kalman Filters, Rao-Blackwellised Particle Filters, and maximum likelihood estimation.
    - delineated the basic challenges connected to efficiency and robust data association.
- Algorithmic analysis age: 2004-2015
    - study of fundamental properties of SLAM, including observability, convergence, and consistency.
    - the key role of sparsity towards efficient SLAM solvers was also understood, and the main open-source SLAM libraries were developed.