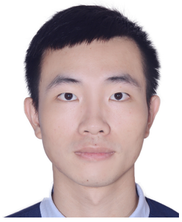
Learn to Calibrate

Perception and Learning for Robotics
Project Proposal
Supervised by: Florian Tschopp, Andrei Cramariuc, Michel Breyer
March 8, 2020

GROUP MEMBERS

Le Chen Yunke Ao




I. DESCRIPTION OF THE PROJECT

Since visual-inertial systems have been prevailed in a wide range of applications [1], precise calibration is of great importance. Typically, it requires performing sophisticated motion primitives in front of a calibration target [2]. In this project, instead of performing this task manually and empirically, our goal is to apply reinforcement learning to learn the best motion primitives for enough calibration precision. With this result, we aim to achieve automatic calibration of an arbitrary visual-inertial system using a robotic arm.

II. WORK PACKAGES AND TIMELINE

The visual-inertial sensor system considered in this project consists of multiple cameras and an IMU. The calibration parameters of such systems include camera intrinsics, IMU intrinsics, multi-cameras extrinsic, the transformation and the temporal offset between the cameras and IMU [2]. Calibrating those parameters requires a wide variety of poses and enough excitation, making the routine complicated to meet the accuracy requirement. Instead of performing calibration by hand or with a manually programmed robotic arm, we want to integrate reinforcement learning to learn the best motion primitives and perform them on a robotic arm.

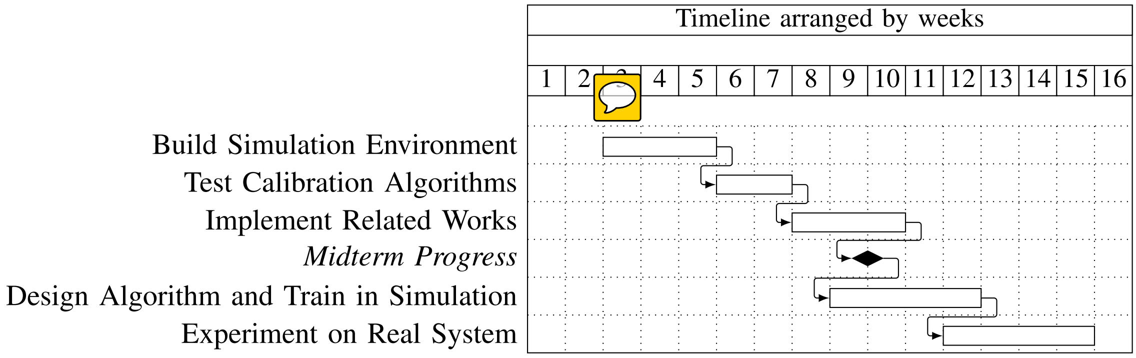
Firstly, we are going to choose appropriate models for cameras and IMU, which determine the parameters to be identified. We also introduce the formalism of the probabilistic SLAM, formalizing the calibration problem in factor graph representation that contains visual-inertial keyframe states, landmarks and calibration states. Then we need to interpret the calibration problem as an MDP or POMDP, for which we implement reinforcement learning to learn the best calibration policy.

Then we will build a simulation environment which is necessary for training and testing the algorithms before implementation on real platforms. It is used to simulate the calibration process of different visual-inertial systems. As for environment settings, the intrinsics of the visual-inertial system and the pattern of the marker should be adjustable. It should also allow the motion of the sensor system to be controlled and the data of sensors to be gathered by other programs. We plan to build this simulator with Gazebo.

After the simulation environment is built, we will apply different algorithms in this simulation. At first, the state-of-the-art calibration methods (such as Kalibr [3]) would be implemented, where the motion is chosen with some empirical trajectories. Then, we will reimplement observability-aware trajectory optimization [4] and Q-learning [5] to see how the performance will improve with these previous related works. Next, we will design our reinforcement learning (RL) algorithm that could cope with long-term goals such as the number of steps of motion apart from observability, and we may further apply information theory to analyze the observability [6]. The challenge in this part lies in designing an algorithm that can achieve all targets. We could either include everything in a sophisticated reward design or treat targets differently (such as constrains) and apply more powerful algorithms.

We will first train our algorithm in the simulation. The challenge here is training RL algorithms using the gazebo simulation. If we use on-policy algorithms, it would introduce high computation complexity to interact and learn in real-time with low sample efficiency. One possible way to tackle this problem would be training with off-policy data with off-policy algorithms, but it also requires the behavior policies to be reasonable policies. We plan to implement calibration in C++ and reinforcement learning algorithms in python.

Finally, our method will also be applied on real visual-inertial systems such as VersaVIS [7] or Intel Realsense d435i mounted on a Franka Emika Panda robot arm. In the simulation, as there is no constraint on the movement of the visual-inertial system, it can move arbitrarily, which is not possible in real robot systems. Since the Franka Emika Panda robot arm has 7 degrees of freedom, the design of motion control policy satisfying the calibration requirements would be a challenge. Besides, due to the demand for a large quantity of data to learn a good policy with RL, in most cases, it is unrealistic to train using real platforms. However, the simulation environment and real environment are still quite different, therefore learning a practical policy for real platform with simulation data (sim to real) is also a challenge in this part.



III. Outcomes and Demonstration

The evaluation experiment will be conducted using simulated as well as real-world data. For the simulation, we will generate noisy synthetic data to derive the optimal policy and estimate the calibration parameters. Since we know the ground-truth values for the parameters, the simulation allows for comparisons between our RL-based method and other methods such as trajectory optimization. The expected outcome is that our calibration results could converge to within 3σ of the ground-truth (or reach a fully calibrated state) in fewer steps of movement compared with other algorithms. For the real-world experiment, there are two potential ways to evaluate the calibrations since the ground-truth values are not available. Firstly, we can use the accuracy of the motion estimation (based on our calibrations) against the measurement from the motion-capture system as the main evaluation metric. Secondly, we can use the evaluation method of hand-eye calibration. The displacement between the target that the robot aims to reach and the real position of the end-effector can be treated as the evaluation metric. With the same number of steps of motion, our method is expected to result in higher accuracy. We may present a video demonstrating the calibration method as well as the experiment at the end of the semester.

References

[1] Guoquan Huang. Visual-inertial navigation: A concise review. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 9572–9582. IEEE, 2019.

[2] Paul Furgale, Joern Rehder, and Roland Siegwart. Unified temporal and spatial calibration for multi-sensor systems. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1280–1286. IEEE, 2013.

[3] Joern Rehder, Janosch Nikolic, Thomas Schneider, Timo Hinzmann, and Roland Siegwart. Extending kalibr: Calibrating the extrinsics of multiple imus and of individual axes. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4304–4311. IEEE, 2016.

[4] Karol Hausman, James Preiss, Gaurav S Sukhatme, and Stephan Weiss. Observability-aware trajectory optimization for self-calibration with application to uavs. *IEEE Robotics and Automation Letters*, 2(3):1770–1777, 2017.

[5] Fernando Nobre and Christoffer Heckman. Learning to calibrate: Reinforcement learning for guided calibration of visual–inertial rigs. *The International Journal of Robotics Research*, 38(12-13):1388–1402, 2019.

[6] Thomas Schneider, Mingyang Li, Cesar Cadena, Juan Nieto, and Roland Siegwart. Observability-aware self-calibration of visual and inertial sensors for ego-motion estimation. *IEEE Sensors Journal*, 19(10):3846–3860, 2019.

[7] Florian Tschopp, Michael Riner, Marius Fehr, Lukas Bernreiter, Fadri Furrer, Tonci Novkovic, Andreas Pfrunder, Cesar Cadena, Roland Siegwart, and Juan Nieto. Versavis: An open versatile multi-camera visual-inertial sensor suite. *arXiv preprint arXiv:1912.02469*, 2019.