Progress Report

Bardia Mojra

February 20, 2023

Robotic Vision Lab

The University of Texas at Arlington

1 Progress

The following items are listed in the order of priority:

- DLO Dataset (IROS - March 1st.): Last night, I was able to test the RealSense+UR5e data collection system in operation mode. I need to disable the watchdog timer, and UR5e will run properly, but I rather keep it or replace it with a similar safety feature. The code is developed based on the latest versions of OpenCV, RealSense, and UR5e Python API, making it useful for further development for at least five years. Moreover, the code is objected-oriented and coding-standard compliant, making it portable, modular, and easy to integrate into various applications. It can easily support multiple dataset papers. rsDevice module is the object-oriented implementation of RealSense API that currently supports RGBD capture, alignment, depth clipping, decimation, filtering, save-to-file, live 2D view, point cloud generation, bag file parser (rsParser Module), and more. The implementation supports expansion to all available features provided by the manufacturer. While working on the rsDevice module, I learned the original C++ implementation of the RealSense API is buggy and requires special care while handling. C++ does not have built-in garbage collection at the compiler level nor it provides routines such as Malloc() and Calloc() for manual garbage collection. These routines were introduced in C11 version of the C language, but I can not recall the same for C++. Ultimately, their solution is to run the driver (pipeline) at the Kernel level, similar to ROS and in Docker, as everything is contained. My solution was to declare RealSense native objects as global objects within the rsDevice module. The parser must be a separate module because once the pipeline is closed, the deleted RealSense objects linger and do not allow a new pipeline to start. Moreover, OpenCV (live-view mode) interferes with bag file write operation; recorded bag files will have corrupted indexes.

- Maicol:

- DLO Manipulation (IROS): [1].

- XEst (RAL —): No update.

2 Research Plan

This section outlines my current research plan for the next 3 months, 6 months, and 1 year. Moreover, I have included open projects and ideas to keep track of them.
Target conferences: ICRA, IROS (March), CASE (Late Feb.), NIPS.
Target Journals: RAL, CVPR, CORAL.

2.1 Research Plan:

- **3 months:** The primary objective will be to publish the DLO dataset paper, (DLO-1), finished my classes, and to meet my next Ph.D. milestone, the comprehensive exam. My goal is to submit the DLO dataset paper to IROS by March 1st.
- **6 months:** Next, I want to explore using DMD as a method to retrieve the correct Quaternion solution for the QuEst method, (QuEst-01). I believe this testing this is fairly fast and I should be able to publish that paper fairly quickly. I believe the RAL would be an appropriate journal to target; we can discuss this further with Dr. Gans to get his input.
- **1 year:** Next, I want to focus on (PIKO-01) as a method for fast online system identification. My aim is to confirm this method by comparing it against existing Koopman-based methods. In the following work, I will extend this method to control DLOs in real time (DLO-02).

2.2 Research Pipeline:

- DLO-01 (IROS - March 1st, 2023): DLO manipulation dataset with DLO configuration and gripper pose, as well as the gripper control input. Ideally, UR5 back-EMF current and bus voltage should be recorded. A DLO mount is introduced. A method for configuration estimation is introduced. Perhaps, a method for learning DLO dynamic can be trained and introduced.

- QuEst-01 (IROS): Optimal transform solution for QuEst based on dominant mode decomposition (DMD).
- PIKO-01 (TBD): This work leverages DMD and Physics-Informed machine learning to extract low-dimensional coherent modal structures from dynamic data. This method will extend DMD-based approaches to include mixed basis functions. Moreover, this method will automatically try to find the best fit at a specified range of ranks. This method will be validated by comparing it against the existing Koopman-based MPC control schemes for VTOL-DIP method and introducing a method for controlling VTOL-TIP in simulation. This method will become the backbone of my Koopman-based MPC control research effort.
- DLO-02 (TBD): This method will extend PIKO-01 to a control method for the DLO-01 dataset.

References

[1] I. Abraham, G. De La Torre, and T. D. Murphey, "Model-based control using koopman operators," *arXiv preprint arXiv:1709.01568*, 2017.