Progress Report

Bardia Mojra

July 11, 2022

Robotic Vision Lab

The University of Texas at Arlington

1 Specific Research Goals

- VPQEKF (May 30th): Work on the paper.
- DLO Manipulation Dataset (ICRA - Sept. 1st)

2 To Do

- QEKF Paper - 30% extension (June 30th):
- Implementation (May 30th):
  - Noise issue: noise cannot be modeled - revisit
  - Rewrite RQuEst (RANSAC-QuEst) – Done
  - Replace the relative pose estimation routine in SfM example with RQuEst – Done
  - SfM: RQuEst cannot find solution – under investigation
- DLO Manipulation: ICRA - Sept. 1st
  - Work on the paper everyday – up-coming
  - ICRA 2022 RL workshops: gym, stable-baseline3, and RL zoo – on-going
  - Setup digital twin reinforcement learing setup:
    * Unity Robotics extension setup – on-going.
    * Design dynamic DLO data collection system.
    * Build work cell. – on-going
    * Collect data and create a dataset.
    * Define evaluation metrics.
    * Create a high frequency RGBD dataset with UV-frames and open-loop input control actions as the ground truth.
  - Real-Time Preception
    * Deep learning methods for keypoint pose estimation in real-time.
    * Use UV dye dataset
    * Use PVNet-like approach for known-object pose estimation.
  - Learning DLO Dynamics and System Identification

    - List feasible approached for learing DLO dynamics
    - Model dynamics and deformity in a latent space
  - Real-Time Control
    - Time model inference, using auto-encoders generate the lowest dimensional representation for each object.
    - Use another GAN model for object deformity for each object.
    - Evaluate encoded representation for accuracy.
    - Used another GAN to explore other abstraced representations from individual encoded representation. In theory, we can create a low dimensionsal representation for multiple similar objects, given all individual low-dimensional representations. This is inspired by "fundamental principles first" approach which has universal applicability.

3 Progress

The following items are listed in the order of priority:

- XEst (RAL —): I integrated Kaveh's QuEst algorithm into Matlab

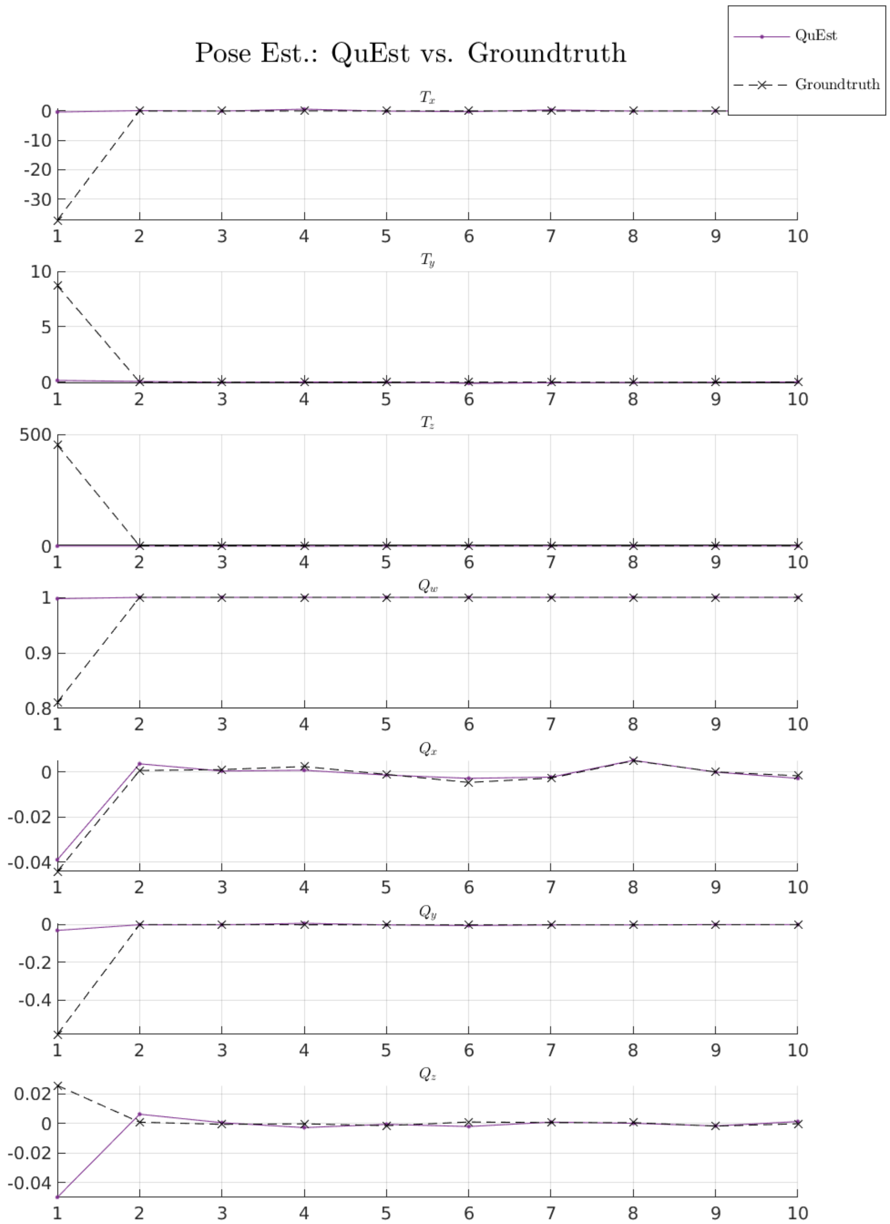

Figure 1: Pose module: Estimated pose log using QuEst.

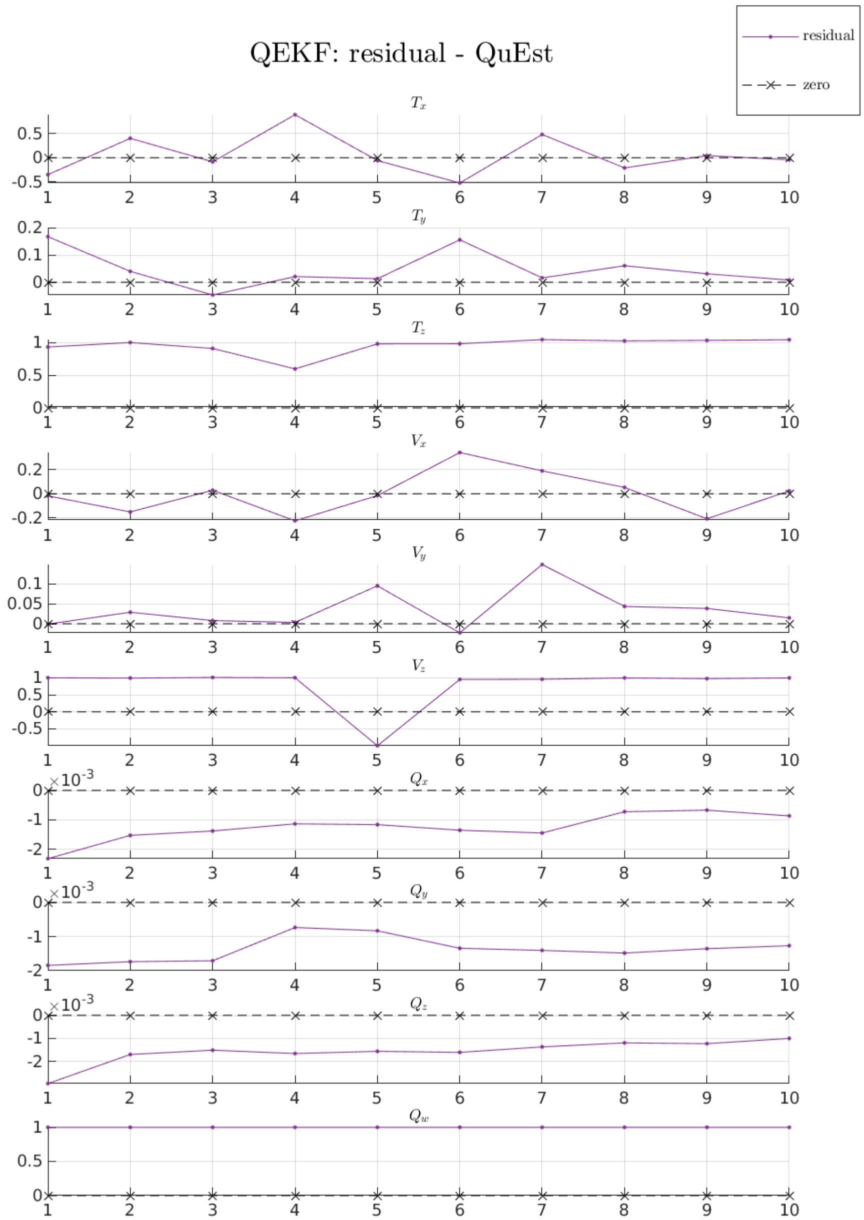

Figure 2: QEKF module: State ($T_{xyz},\ V_{xyz},\ Q_{xyzw}$) residual log with QuEst estimations as state measurements.

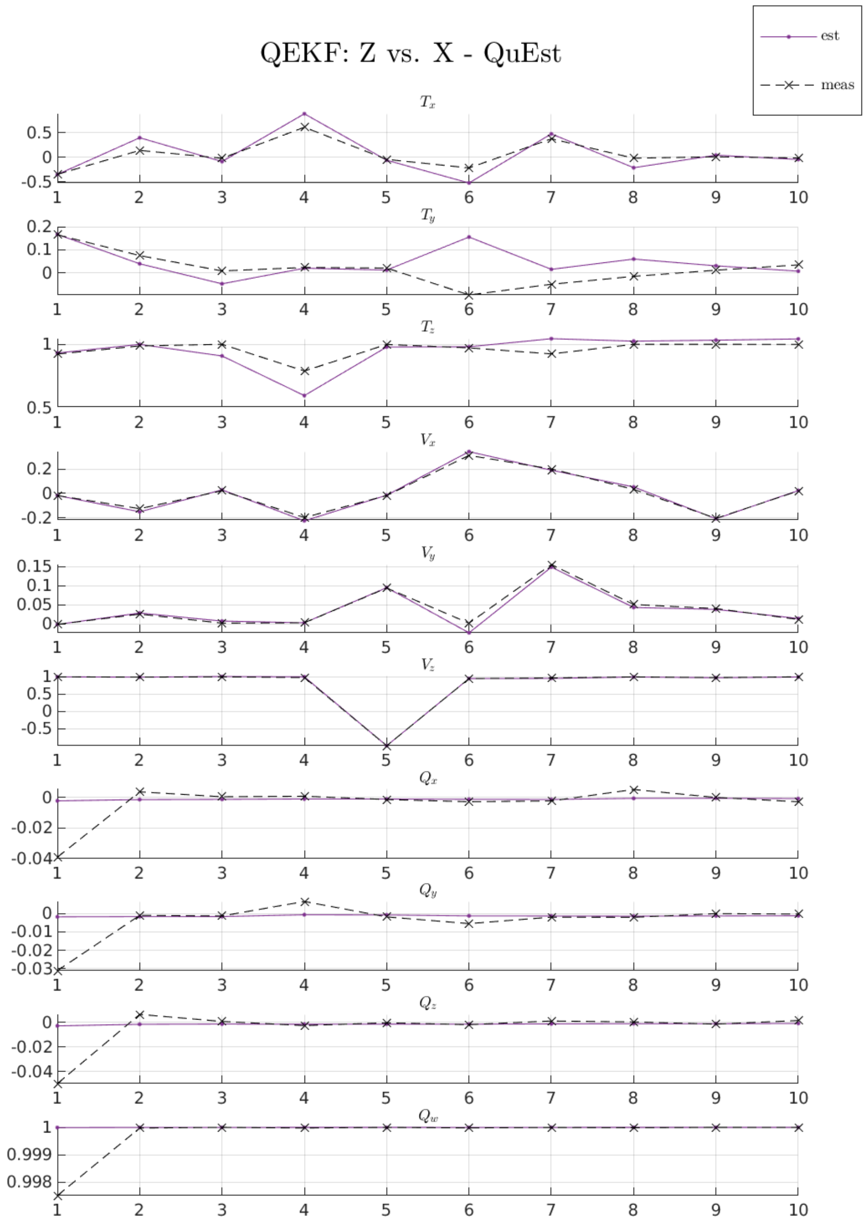

Figure 3: QEKF module: State measurement vs. state estimation for QuEst.

Regarding our direction moving forward, I believe Dr. Gans and I have a made new realization. EKF remains one of the state-of-the-art methods for real-time state estimation with at least partially known dynamic model and completely known navigation environment. But related implementations for VSLAM and Structure from Motion (SfM) rely on *Pose Graph*, *loop closure* via *place recognition*, and two *Bundle Adjustments* (one local and during run-time, the other at post processing). Related work by Nister [1] makes no mention of EKF, loop closure, nor pose graph but based on my understanding I believe they implemented a variation of VSLAM or SfM. I was able to successfully run VSLAM and SfM in Matlab. Figures 4-5, 6-8 show the outputs for VSLAM and SfM, respectively. Right now, I am in the process of rewriting the relative pose estimation algorithm to use QuEst with RANSAC (RQuEst) and triangulation instead of Sampson distance. Triangulation is a more robust method for finding the correct solution as Sampson distance is affected by noise. Moreover, they use pose graph and have a much more advanced method for point feature matching, keyframe selection, and view tracking which I am adopting.

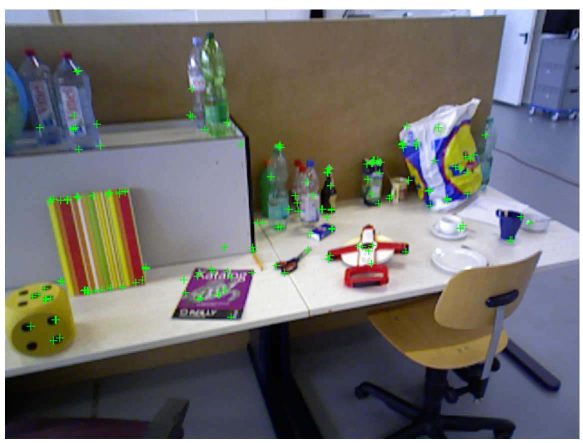

Figure 4: VSLAM: Final frame matched point features.

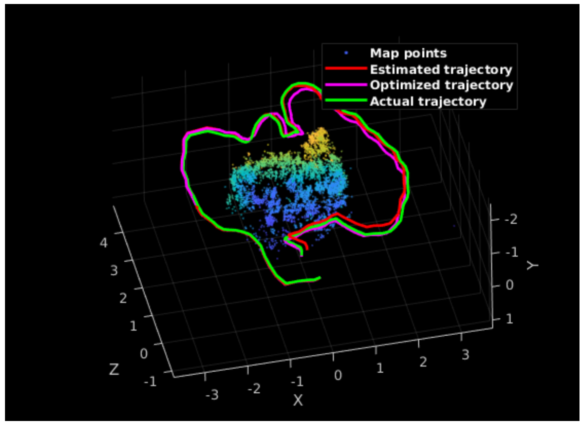

Figure 5: VSLAM: Final track.

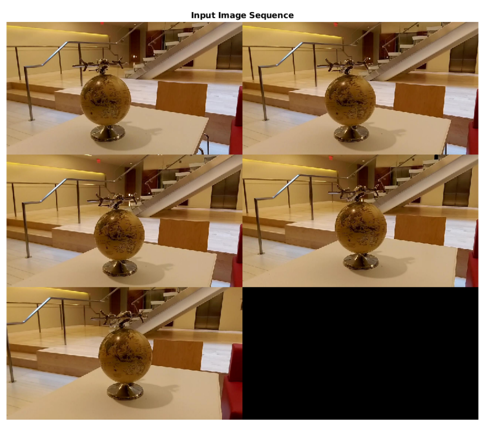

Figure 6: SfM: Input image sequence.

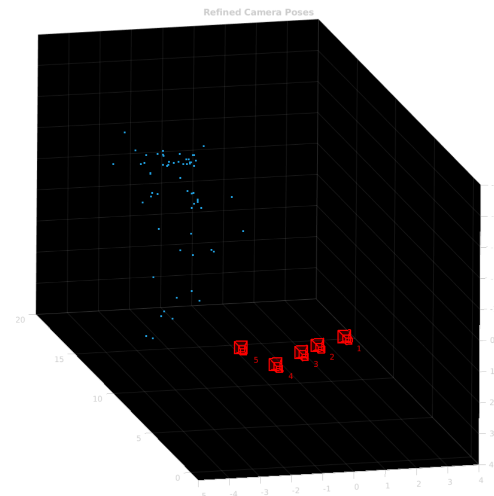

Figure 7: SfM: Refined camera poses.

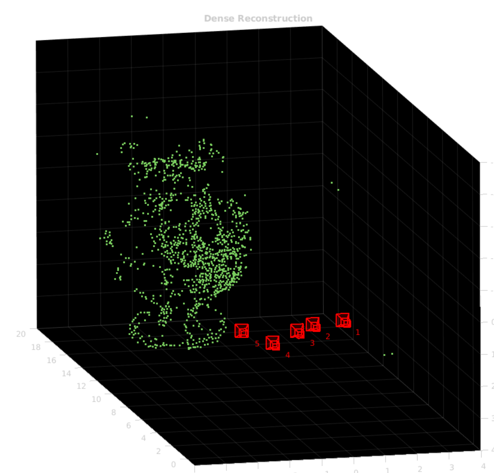

Figure 8: SfM: Dense reconstruction.

- XEst - Semantic segmentation (RAL - April 30st, 2022): No update on implementing [2].
- DLO Dataset: I finished tutorials on Unity and learned basic C# syntax. I also looked into Omniverse and Isaac Sim and decided to switch to that. I will have to install on TACC and asked Linus and Maicol to create their own accounts as well. I learned about a major shortcut that eliminates much of the need for setting up ROS. In one of the views, the presenting Nvidia engineer use Jupyter Noteboot to directly interface with a real robotic arm and synced its digital twin with physical arm which he controlling via real-time input. More
- Linus (REU): He will start on ROS2 tutorials soon.
- Maicol (REU): He will start on TurtleBot tutorials soon.
- Myself (with REU): I will set up Omniverse on TACC and VNC connection this week.
- DLO Control (MuJuCo): No update.
- Grasping Project (DLO-03): I am making this a part of the DLO project.
- PyTorch Tutorials: Transfer learning.
- Manifold learning: Marcus emailed me some papers, I will read them and reply to him. I am not particularly interested in the project but his ideas are interesting and I would like to help him if I can. He is very knowledgeable on mathematics and I cherish that.

4 Intermediate Goals - Fall 2021:

- QEKF: Finish paper.
- UR5e: Do the tutorials.

References

[1] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 6, pp. 756–770, 2004.

[2] I. Ballester, A. Fontan, J. Civera, K. H. Strobl, and R. Triebel, "Dot: dynamic object tracking for visual slam," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11705–11711, IEEE, 2021.