Visual Load Pose Estimation in a Multi-UAV Slung Load System

Harvey Merton

I. INTRODUCTION AND RELATED WORK

From 2010 to 2013, the GRASP Lab at the University of Pennsylvania published a series of papers demonstrating that multiple-drone systems can successfully carry a cable-suspended payload [1], [2], [3], [4]. Reference [5] builds on this work by using the load position and orientation as feedback to a geometric controller. This feedback is usually retrieved through a motion capture system [6], which are only available in indoors.

Vision-based systems for load feedback can work indoors and outdoors. Such a vision-based system was used to provide measurements of a slung load relative to a single carrying helicopter in [7], and a single carrying quadcopter in [8]. These measurements were integrated with known 2 degree of freedom (DOF) pendulum dynamics in an unscented Kalman filter (UKF) and Extended Kalman filter (EKF) respectively. To the author's knowledge, experiments on visual pose feedback in a multi-drone slung load system are yet to be published. Current outdoor systems either use no load feedback in a quasi-static approach, or use sensors (such as an IMU or GPS) mounted directly to the load [9]. This work aims to be the first to use visual feedback to achieve a load pose estimate suitable for a multi-drone slung load system operating outdoors.

II. SYSTEM ARCHITECTURE

The multi-drone slung load system is modelled in the Gazebo dynamics simulator as seen in Fig. 1.

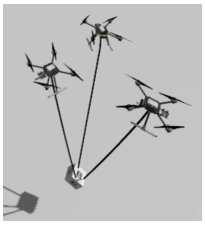

Fig. 1: Multi-drone slung load system in Gazebo simulation.

Appendix I summarises the ROS2 nodes required run a quasi-static formation control procedure to command the drones to lift the load. The code for the vision and estimation modules developed in this project can be found (open-source) here, while video results can be found here.

III. VISUAL LOAD POSE MEASUREMENT

To make the load detection task easier, an Aruco marker is placed on the top face of the load using OpenCV's Aruco module. To stream the images from the simulated cameras to ROS2, the ros_gz_image package is used, while the ros_gz_bridge package is used to get the ground truth pose of the load and drones for evaluating load pose estimates.

The perspective-n-point (PnP) algorithm is designed to solve the pose estimation problem [10]. Mathematically (see equation 1), find the transform T_L^C (where C is the camera frame and L is 'load marker' frame) given the 2D pixel coordinates (u,v), corresponding 3D load coordinates (X_L, Y_L, Z_L), camera intrinsic matrix K and the canonical projection Π.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K}\Pi T_L^C \begin{bmatrix} X_L \\ Y_L \\ Z_L \\ 1 \end{bmatrix} \tag{1}$$

OpenCV implements many PnP variants: IPPE Square [11] (faster for planar marker detection), P3P [12] (minimal solver requiring only 3 points, 4 for no ambiguity) and iterative (uses Levenberg Marquardt [13] to optimize after an initial guess) are used here.

A. Results

The three-drone slung load system is commanded to fly in a circle of radius $2\,m$ at $1.5\,m/s$. The system also rotates at a matching rate such that the same side of the box should always face the center of the circular trajectory. The results at a snapshot in time from a single drone's PnP estimation can be seen in Fig. 2.

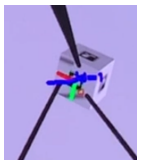

Fig. 2: Cropped image stream from a simulated camera mounted to a drone in the slung load system. The overlayed axes (red = x, green = y, blue = z) represent the estimated marker axes using OpenCV's IPPE Square PnP algorithm.

The position and orientation error over a few trajectory loops can be seen in Fig. 3. These errors are defined by equations 2 and 3 (geodesic distance) respectively.

$$t^C_{L\,error} = t^C_{L\,estimated\,(est)} - t^C_{L\,ground\,truth\,(gt)} \tag{2}$$

$$dist_\theta(R^W_{L,\,est}, R^W_{L,\,gt}) = |arccos(\frac{tr((R^W_{L,\,est})^T R^W_{L,\,gt}) - 1}{2})| \tag{3}$$

Despite many debugging efforts, the translation error remained around $0.6\,m$ on average, while the orientation error remained around $50°$. Appendix II clearly shows that the position error offset is predominantly due to z-error. It also

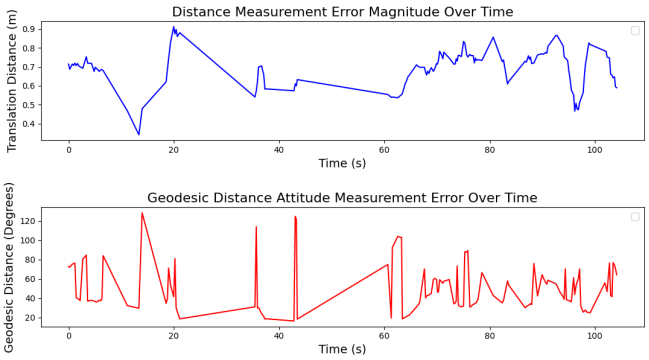

Fig. 3: IPPE square pose estimation errors over circular trajectory.

shows large spikes in yaw and roll error which correspond to a flipping of the z-axis seen visually in the image stream. Debugging attempts included measuring ground truth distances in Gazebo, checking the marker size, intrinsic calibration matrix and the distortion matrix, all to no avail. Further, results from other PnP algorithms (see Appendix III) show similar results suggesting that the PnP algorithm is not failing. It is interesting to note that the P3P algorithm sees high translation error spikes (of around $2\,m$) corresponding to times when the marker points are close together in the image. This suggests that IPPE Square's planar assumption and iterative pnp's optimization are particularly useful when key points are close together in an image.

IV. LOAD STATE ESTIMATION

Kalman filters obtain optimal state estimates by fusing a series of measurements over time with a known dynamics model. The two variants investigated here for load position estimation are the EKF and the UKF (discussed in [14]). The kalmanif library provides a C++ interface for both of these algorithms directly on Lie groups [15].

The key implementation performed for this project is the definition of the measurement equation 4 and the state update equation 5. The measurement equation simply defines the position of drone i relative to the load $y_{d_i}^L$ (the inverse of what is measured by PnP, with static transform offsets) given the position of drone i $t_{d_i}^W$ (known from GPS).

$$\mathbf{y}_{d_i}^L = \begin{bmatrix} \mathbf{R}_L^W & \mathbf{t}_L^W \\ \mathbf{0} & 1 \end{bmatrix}^{-1} \mathbf{t}_{\mathbf{d_i}}^{\mathbf{W}} \tag{4}$$

The state update equation is implemented in a linear form as provided by the kalmanif library. This means that the nonlinear load dynamics (equations 8) derived in [4] have to be captured in the control input term (equation 7) derived from kinematic equations with constant acceleration.

$$\mathbf{X}_L = \mathbf{X}_L + \mathbf{U} \tag{5}$$

where

$$\mathbf{X}_L = \begin{bmatrix} \mathbf{t}_L^W \\ \mathrm{RPY}_{ZYX}\left(\mathbf{R}_L^W\right) \end{bmatrix} = \begin{bmatrix} \mathbf{x}_L \\ \boldsymbol{\theta}_L \end{bmatrix} \tag{6}$$

where RPY_{ZYX} maps a rotation matrix into its roll, pitch and yaw Euler angles in the ZYX convention and

$$\mathbf{U} = d\mathbf{X}_L = \begin{bmatrix} \dot{\mathbf{x}}_L \\ \dot{\boldsymbol{\theta}}_L \end{bmatrix} dt + \frac{1}{2} \begin{bmatrix} \ddot{\mathbf{x}}_L \\ \ddot{\boldsymbol{\theta}}_L \end{bmatrix} dt^2 \tag{7}$$

given

$$\ddot{\mathbf{x}}_L = \frac{1}{m_L}[-\sum \mathbf{R}_L^W T_i \mathbf{q}_i] - g\mathbf{e}_3 \tag{8a}$$

$$\ddot{\theta}_L = \mathbf{J}_L^{-1}[\sum \mathbf{r}_i \times -T_i \mathbf{q}_i - \dot{\theta}_L \times \mathbf{J}_L \dot{\theta}_L] \tag{8b}$$

where $\mathbf{J}_L$ defines the rotational inertia of the load, $\mathbf{r}_i$ is the vector from the load's center of mass to the i'th cable's attachment point on the load and $T_i\mathbf{q}_i$ is the i'th cable's tension vector from the i'th quadrotor to its attachment point on the load.

A. Results

As the tension vectors $T_i\mathbf{q}_i$ are difficult to retrieve from Gazebo, a custom dynamics simulation that implements the full system dynamics is used. The trajectory is the same as described in section III-A, except with no load or drone rotation (only load position is estimated currently).

Fig. 4 presents the trajectories predicted by the filters while Appendix IV presents the corresponding position error magnitude. Both graphs show that the unfiltered trajectory, without measurement corrections, veers away from the ground truth over time while the EKF and UKF average around 0 error for the whole trajectory. It is interesting to note that the trajectories predicted by the EKF and UKF are exactly the same currently, due to the linearlized system dynamics implemented in equation 5.

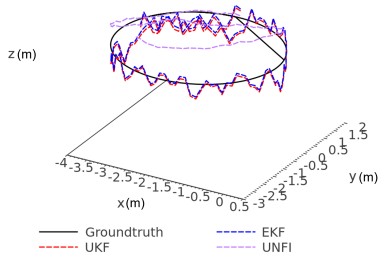

Fig. 4: EKF and UKF predicted trajectories vs the unfiltered (UNFI) trajectory obtained without measurement correction. Measurement noise is modelled as $\mathbf{y_{noise}} \sim \mathcal{N}(0, 0.01)$ while process noise is modelled with control input uncertainty $\mathbf{u_{translation\ noise}} \sim \mathcal{N}(0, 1e^{-4})$, $\mathbf{u_{rotation\ noise}} \sim \mathcal{N}(0, 9e^{-6})$.

V. CONCLUSIONS AND FUTURE WORK

Using a vision measurement system coupled with an EKF or UKF for load pose estimation is feasible in a multi-drone slung load system. Future work for the measurement system involves debugging the source of the predicted translation offset and reducing the noise on the orientation estimate when the z-axis flips. For the state estimators, capturing the nonlinearities directly in the update equation rather than the control input should give different results for the EKF and UKF. The intention is also to obtain a load orientation estimate and tune the system and measurement noise covariances.

REFERENCES

[1] J. Fink, N. Michael, S. Kim, and V. Kumar, "Planning and control for cooperative manipulation and transportation with aerial robots," *The International Journal of Robotics Research*, vol. 30, no. 3, pp. 324–334, Mar. 2011. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0278364910382803

[2] N. Michael, J. Fink, and V. Kumar, "Cooperative manipulation and transportation with aerial robots," *Autonomous Robots*, vol. 30, no. 1, pp. 73–86, Jan. 2011. [Online]. Available: http://link.springer.com/10.1007/s10514-010-9205-0

[3] K. Sreenath, N. Michael, and V. Kumar, "Trajectory generation and control of a quadrotor with a cable-suspended load - A differentially-flat hybrid system," in *2013 IEEE International Conference on Robotics and Automation*. Karlsruhe, Germany: IEEE, May 2013, pp. 4888–4895. [Online]. Available: http://ieeexplore.ieee.org/document/6631275/

[4] K. Sreenath and V. Kumar, "Dynamics, Control and Planning for Cooperative Manipulation of Payloads Suspended by Cables from Multiple Quadrotor Robots," in *Robotics: Science and Systems IX*. Robotics: Science and Systems Foundation, Jun. 2013. [Online]. Available: http://www.roboticsproceedings.org/rss09/p11.pdf

[5] G. Wu and K. Sreenath, "Geometric control of multiple quadrotors transporting a rigid-body load," in *53rd IEEE Conference on Decision and Control*. Los Angeles, CA, USA: IEEE, Dec. 2014, pp. 6141–6148. [Online]. Available: http://ieeexplore.ieee.org/document/7040351/

[6] S. Tang, K. Sreenath, and V. Kumar, "Multi-robot Trajectory Generation for an Aerial Payload Transport System," in *Robotics Research*, N. M. Amato, G. Hager, S. Thomas, and M. Torres-Torriti, Eds. Cham: Springer International Publishing, 2020, vol. 10, pp. 1055–1071, series Title: Springer Proceedings in Advanced Robotics. [Online]. Available: http://link.springer.com/10.1007/978-3-030-28619-4_70

[7] M. Bisgaard, A. La Cour-Harbo, E. N. Johnson, and J. D. Bendtsen, "VISION AIDED STATE ESTIMATOR FOR HELICOPTER SLUNG LOAD SYSTEM," *IFAC Proceedings Volumes*, vol. 40, no. 7, pp. 425–430, 2007. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1474667015332791

[8] S. Tang, V. Wuest, and V. Kumar, "Aggressive Flight With Suspended Payloads Using Vision-Based Control," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1152–1159, Apr. 2018. [Online]. Available: http://ieeexplore.ieee.org/document/8258883/

[9] E. Bulka, C. He, J. Wehbeh, and I. Sharf, "Experiments on Collaborative Transport of Cable-suspended Payload with Quadrotor UAVs," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*. Dubrovnik, Croatia: IEEE, Jun. 2022, pp. 1465–1473. [Online]. Available: https://ieeexplore.ieee.org/document/9836163/

[10] E. Marchand, H. Uchiyama, and F. Spindler, "Pose Estimation for Augmented Reality: A Hands-On Survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 12, pp. 2633–2651, Dec. 2016. [Online]. Available: http://ieeexplore.ieee.org/document/7368948/

[11] T. Collins and A. Bartoli, "Infinitesimal Plane-Based Pose Estimation," *International Journal of Computer Vision*, vol. 109, no. 3, pp. 252–286, Sep. 2014. [Online]. Available: http://link.springer.com/10.1007/s11263-014-0725-5

[12] Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng, "Complete solution classification for the perspective-three-point problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 930–943, Aug. 2003. [Online]. Available: http://ieeexplore.ieee.org/document/1217599/

[13] E. Eade, "Gauss-Newton / Levenberg-Marquardt Optimization," 2013.

[14] S. J. Julier and J. K. Uhlmann, "A New Extension of the Kalman Filter to Nonlinear Systems," 1997.

[15] J. Solà, J. Deray, and D. Atchuthan, "A micro Lie theory for state estimation in robotics," Dec. 2021, arXiv:1812.01537 [cs]. [Online]. Available: http://arxiv.org/abs/1812.01537

APPENDIX I
SYSTEM ARCHITECTURE

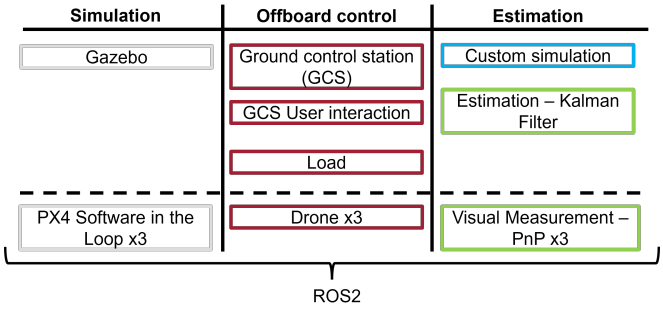

Fig. 5: Modules in the full system architecture linked together by ROS2. Red indicates custom modules written in python, green indicates C++, blue indicates Julia while grey boxes indicate non-custom systems.

APPENDIX II
IPPE SQUARE ERROR COMPONENTS

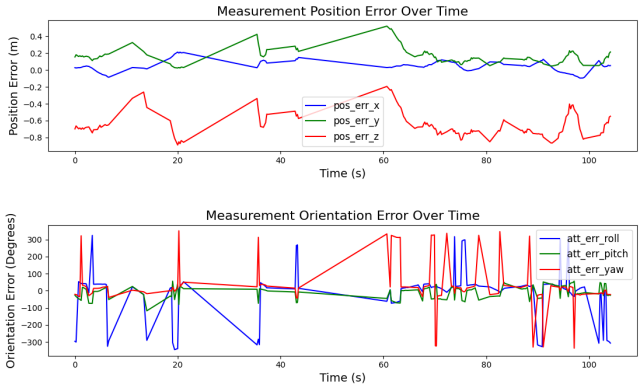

Fig. 6: Components of position and orientation estimation errors over circular trajectory.

APPENDIX III
POSE ESTIMATION PERFORMANCE OF OTHER PNP ALGORITHMS

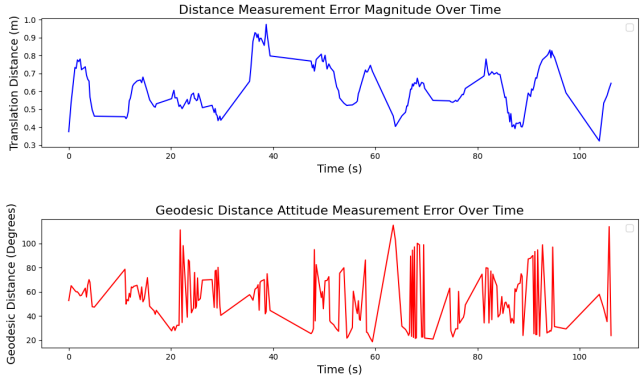

Fig. 7: PnP iterative pose estimation errors over circular trajectory.

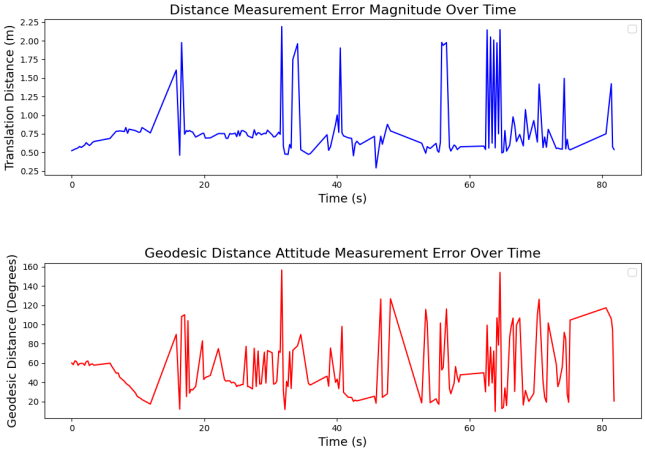

Fig. 8: P3P pose estimation errors over circular trajectory.

APPENDIX IV
KALMAN FILTER POSITION ERROR MAGNITUDE

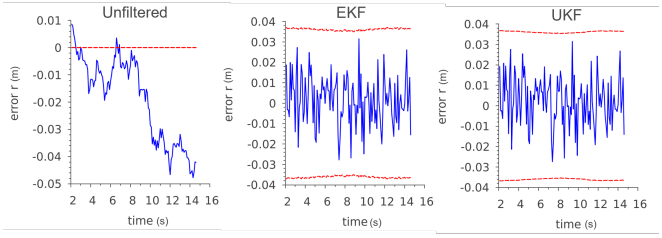

Fig. 9: Predicted pose error magnitude over the trajectory seen in Fig. 4. Blue lines correspond to the mean while the red dotted lines corresponds to 3 standard derivations from the mean (in the filter cases).