Perspective-N-Point Observation Model

Ashwin Disa
Robotics Engineering
Worcester Polytechnic Institute
Email: amdisa@wpi.edu

I. Introduction

Accurate pose estimation is a fundamental requirement in robotics, especially for autonomous aerial systems navigating through known environments. This project focuses on implementing a computer vision-based observation model using the Perspective-n-Point (PnP) method to estimate the pose of a Nano+ quadrotor. The resulting observation model is intended to serve as the measurement model for a subsequent nonlinear filtering project, such as a Particle Filter.

The data used in this project were collected from a downward-facing camera mounted on the drone as it was moved through a trajectory above a mat populated with AprilTags. The AprilTags are arranged in a well-defined 12 × 9 grid with known inter-tag distances, providing a robust reference map in the world frame. This spatial configuration allows us to establish precise 3D-2D correspondences necessary for solving the PnP problem.

The camera calibration parameters, including the intrinsic matrix and distortion coefficients, are provided and are critical for accurate reprojection of world points onto the image plane. Additionally, the relative transformation between the camera and drone body frames must be accounted for to express pose estimates in the robot frame. Ground truth data collected using a motion capture system (Vicon) is available for evaluation but not directly used during estimation.

This project involves several key tasks: (1) developing the pose estimation function based on the PnP method, (2) visualizing and analyzing the estimated drone trajectory compared to ground truth, and (3) quantifying the reliability of the observation model through covariance estimation. The ultimate goal is to create a reliable and reusable module for vision-based localization that integrates seamlessly into future state estimation pipelines.

II. Methodology

This section outlines the implementation details of the Perspective-n-Point (PnP) based observation model used to estimate the pose of a Nano+ quadrotor. The methodology consists of preprocessing camera and tag map data, pose estimation using the PnP algorithm, visualization of the estimated trajectory against ground truth, and estimation of the measurement covariance for use in subsequent nonlinear filtering.

A. Camera Calibration and AprilTag Map Parsing

The camera's intrinsic parameters were extracted from the `parameters.txt` file. These included the camera matrix $\mathbf{K}$ and distortion coefficients. The camera matrix is given as:

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 314.1779 & 0 & 199.4848 \\ 0 & 314.2218 & 113.7838 \\ 0 & 0 & 1 \end{bmatrix}$$

The radial and tangential distortion coefficients are:

$$\text{Radial: } k_1 = -0.438607, \quad k_2 = 0.248625, \quad k_3 = -0.0911$$

$$\text{Tangential: } p_1 = 0.00072, \quad p_2 = -0.000476$$

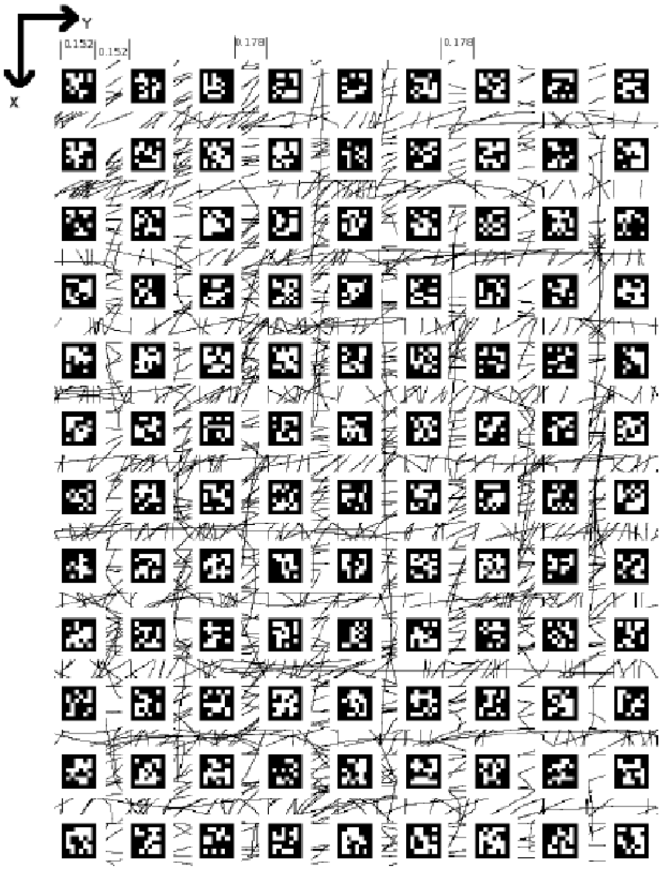

Fig. 1: AprilTag map

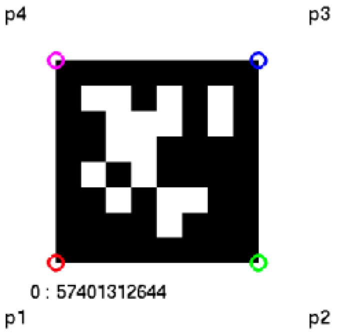

Fig. 2: AprilTag example

AprilTags are arranged in a 12 × 9 grid with a default spacing of 0.152 m. Exceptions occur between columns 3–4 and 6–7, where spacing is 0.178 m. The map origin is defined at the top-left corner of the top-left tag, with x increasing down and y increasing right.

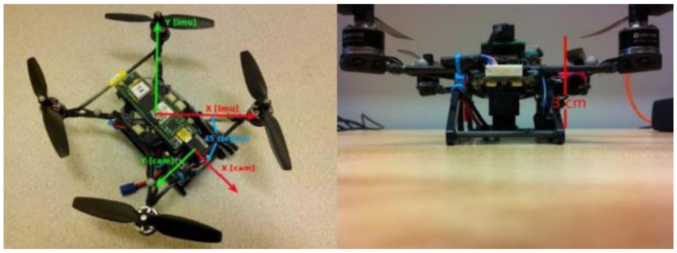

Fig. 3: Camera to drone frame transformation

The extrinsic transformation from the camera frame to the drone body frame is defined by:

$$\mathbf{t}_{\text{cam}\to\text{drone}} = \begin{bmatrix} -0.04 \\ 0 \\ -0.03 \end{bmatrix}, \quad \mathbf{R}_{\text{cam}\to\text{drone}} = \mathbf{R}_{\text{xyz}}(-\pi, 0, -\pi/4)$$

B. Pose Estimation via Perspective-n-Point

The core of the observation model involves solving the PnP problem, which estimates the pose of a calibrated camera given n 3D–2D point correspondences. Each detected AprilTag provides four such correspondences via its corners.

Given:

- 3D world points $\{\mathbf{X}_i\}$ from the tag map.
- 2D image projections $\{\mathbf{x}_i\}$ from detected tag corners.

The PnP algorithm estimates a rotation vector $\mathbf{r}$ and translation vector $\mathbf{t}$ such that:

$$\mathbf{x}_i \sim \pi(\mathbf{K}[\mathbf{R} \mid \mathbf{t}]\mathbf{X}_i)$$

where π denotes the projection operation and $\mathbf{R}$ is obtained from $\mathbf{r}$ using the Rodrigues formula. The camera position in the world frame is:

$$\mathbf{p}_{\text{cam}} = -\mathbf{R}^\top \mathbf{t}$$

The drone position is then computed by applying the known extrinsic transform:

$$\mathbf{p}_{\text{drone}} = \mathbf{p}_{\text{cam}} - \mathbf{R}^\top \mathbf{R}_{\text{cam}\to\text{drone}}^\top \mathbf{t}_{\text{cam}\to\text{drone}}$$

The resulting $\mathbf{p}_{\text{drone}}$ and $\mathbf{R}$ form the estimated pose for each frame.

C. Ground Truth Alignment

Ground truth pose data from a Vicon motion capture system is provided at a higher frequency (100 Hz). For each estimated timestamp, the closest ground truth timestamp is identified using binary search. The aligned ground truth and estimated poses are then used for trajectory comparison and error analysis.

D. Visualization

To evaluate the observation model, we generate a 3D plot visualizing both the estimated and ground truth trajectories. Each pose is represented using 3D orientation axes. Additionally, we compute the MSE in position at each timestep:

$$e_t = \|\mathbf{p}_{\text{est}}(t) - \mathbf{p}_{\text{gt}}(t)\|_2$$

We also compute the mean positional error across all valid timesteps:

$$\bar{e} = \frac{1}{N} \sum_{t=1}^{N} e_t$$

E. Covariance Estimation

To prepare for integrating the observation model into a Bayesian filter (e.g., Kalman or Particle Filter), we estimate the measurement noise covariance matrix. Let the residual vector at time t be:

$$\boldsymbol{\nu}_t = \mathbf{z}_{\text{gt}}(t) - \mathbf{z}_{\text{est}}(t)$$

where $\mathbf{z}$ includes both position and orientation (Euler angles) as:

$$\mathbf{z}(t) = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ \phi(t) \\ \theta(t) \\ \psi(t) \end{bmatrix}$$

Assuming zero-mean Gaussian noise, the sample covariance matrix $\mathbf{R}$ is estimated as:

$$\mathbf{R} = \frac{1}{N-1} \sum_{t=1}^{N} \boldsymbol{\nu}_t \boldsymbol{\nu}_t^\top$$

This covariance matrix $\mathbf{R}$ represents the uncertainty associated with the PnP-based measurement model and will be used as the observation noise model in the subsequent filtering module.

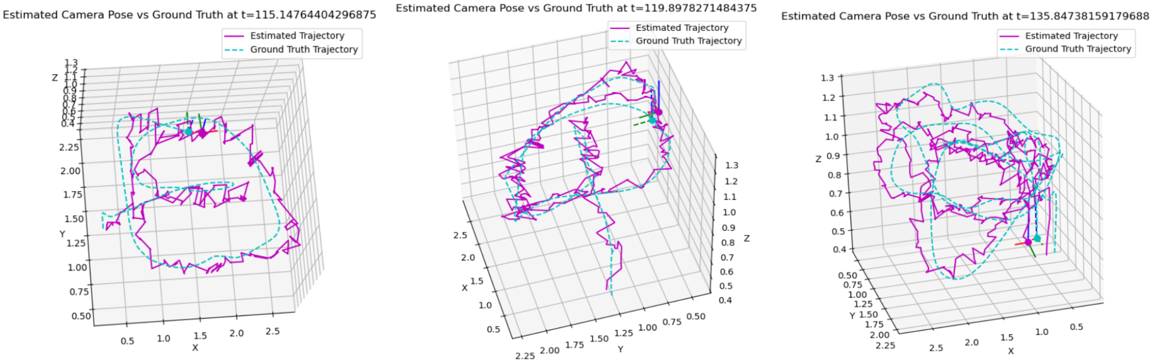

Fig. 4: Estimated and Ground truth trajectory comparison at different instances.

F. Results

This section presents the quantitative evaluation of the PnP-based observation model by comparing the estimated drone poses against the ground truth from a motion capture system. The results include both the estimated covariance of the measurement model and the positional accuracy in terms of Mean Squared Error (MSE).

G. Measurement Noise Covariance

To characterize the uncertainty in the observation model, we estimated the sample covariance matrix $\mathbf{R}$ based on the residuals between the estimated and ground truth poses (position and orientation). The computed 6×6 covariance matrix $\mathbf{R}$ is:

$$\begin{bmatrix} 0.0047 & 0.0004 & -0.0001 & -0.0140 & 0.0041 & 0.0000 \\ 0.0004 & 0.0049 & -0.0003 & -0.1479 & 0.0003 & -0.0001 \\ -0.0001 & -0.0003 & 0.0006 & 0.0059 & -0.0000 & -0.0000 \\ -0.0140 & -0.1479 & 0.0059 & 9.3872 & -0.0180 & 0.0021 \\ 0.0041 & 0.0003 & -0.0000 & -0.0180 & 0.0047 & 0.0001 \\ 0.0000 & -0.0001 & -0.0000 & 0.0021 & 0.0001 & 0.0000 \end{bmatrix}$$

This matrix captures the variance and correlation of errors in both the position (x, y, z) and orientation (ϕ, θ, ψ) components of the estimated pose. Notably, the diagonal elements indicate that the largest variance is observed in the yaw angle, which is consistent with the expectation that orientation estimation from planar tags may suffer in low-texture or ambiguous viewing angles.

H. Trajectory Accuracy

The positional accuracy of the pose estimation algorithm was evaluated using the Mean Squared Error (MSE) between the estimated and ground truth positions over all valid frames. The MSE is defined as:

$$\text{MSE} = \frac{1}{N} \sum_{t=1}^{N} \|\mathbf{p}_{\text{est}}(t) - \mathbf{p}_{\text{gt}}(t)\|_2^2$$

The resulting MSE in position is:

$$\text{Mean Squared Error} = 0.1781$$

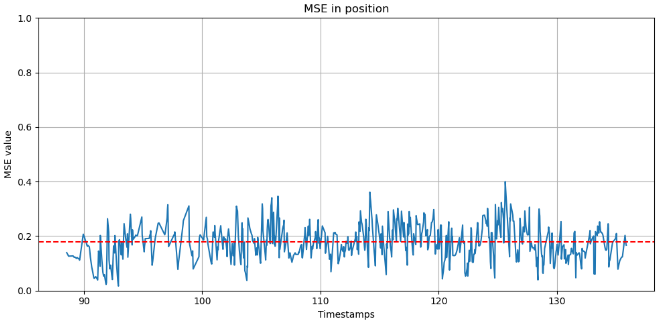

Fig. 5: Mean Squared Error in position

This result indicates a reasonably accurate position estimation from the vision-based system, given the limitations of the AprilTag field and camera resolution. The error magnitude is within acceptable bounds for use in subsequent nonlinear filtering pipelines such as the Particle Filter or the Extended Kalman Filter.