SPAICE Technical Challenge: 3D Tracking of a Single Drone

Nicholas Morrison
October 2025

0. Introduction

This report proposes a multi-camera vision system to detect and track a single flying drone in real-time. The pipeline uses YOLO for 2D detection, triangulation for depth estimation, and a Kalman filter for temporal tracking. The goal is to provide accurate 3D localization of the drone across complex trajectories, while remaining efficient and robust to challenges like occlusion and background clutter. Further extensions including thermal/RGB fusion and non-visual sensor integration are also discussed. A proof-of-concept implementation is demonstrated using the ETH multi-view drone dataset, with code available on Github.

1. System Design Overview

The proposed system uses a network of fixed, monocular RGB cameras positioned around the monitored area. A wide-angle camera, such as a GoPro, is placed centrally to maximize coverage and ensure that the drone remains within the field of view. Additional cameras with standard lenses are distributed at various locations to provide high-quality video feeds with overlapping perspectives, which are necessary for triangulation of the drone's 3D positions. The cameras remain static rather than mounted on gimbals to simplify the geometry. While gimbaled systems could help maintain the target in view, they are less suitable for tracking fast, erratic motions and can introduce additional complexity to calibration.

When designing this pipeline, several modalities were considered for depth sensing. Stereo rigs, which function like human eyes, can estimate depth through parallax. However, their accuracy deteriorates at long ranges due to small disparities. RGB-D sensors, which allow active depth sensing through time-of-flight or structured light, are highly effective for indoor settings but often fail outside. Hyperspectral or multispectral sensors are powerful, but expensive and slow for real-time deployment. For this drone monitoring scenario, a multi-camera RGB network remains the most practical and low latency solution. In addition to the camera network, strong tracking algorithms help to bridge the gaps when detections are lost.

Calibration is crucial to the system's accuracy. Each camera's intrinsic parameters are obtained using standard checkerboard calibration, which allows estimation of focal length, principal point, and distortion coefficients. This step is particularly important when using wide-angle lenses like GoPro's, where fisheye distortion must be corrected. Extrinsic calibration can be derived from total-station measurements. Relative poses between the cameras are established in a common coordinate frame from these known positions and rotations. Finally, hardware synchronization triggers all cameras simultaneously so that each frame represents the same instant in time. For tracking, certain frames can be skipped to reduce latency.

2. Algorithmic Approach

The pipeline consists of three main stages: detection, 3D position estimation, and tracking. For this process, I relied on a combination of traditional foundation models from university classes and further research to determine which state-of-the-art methods are currently used in industry.

2.1 Detection

The system uses YOLOv8, a lightweight CNN detector from Ultralytics, which offers a nano model that works well on edge devices. YOLO is well-suited for this application as it achieves real-time inference speeds, while maintaining performance on small objects by learning general representations. The neural network predicts bounding boxes and class probabilities directly from the whole image in a single-shot evaluation. YOLO can be finetuned on flying drone data to increase performance. Traditional alternatives, such as frame differencing or optical flow, struggle with dynamic outdoor environments like moving clouds. However, they may be useful to supplement detection at long ranges when YOLO fails. Other simple approaches like patch similarity or correlation filters are easy to implement, but lack robustness to rotation and occlusion. On the other end of the spectrum, more advanced transformer-based architectures, such as DETR, can be more accurate with unseen drone types, but are computationally heavy for edge deployment. YOLO therefore provides the best balance between speed and accuracy.

2.2 Depth Estimation

Once the drone is detected in at least two camera views, its 3D position can be estimated using epipolar geometry. With the camera intrinsics and extrinsics known, each 2D detection defines a projection ray in space, originating from the camera center and extending through the pixel coordinates. Triangulation estimates the drone's 3D position by solving for the point of closest approach between these rays. Since noise prevents exact intersection, the solution is obtained using least squares or singular value decomposition. When multiple cameras observe the drone simultaneously, the reconstruction becomes more robust. Methods such as bundle adjustment can refine accuracy by optimizing across all cameras jointly.

2.3 Tracking

The estimated 3D points are then passed through a Kalman filter for temporal tracking. The Kalman filter provides a lightweight and efficient means of maintaining a stable trajectory, even when detections are noisy or occasionally missing. It predicts the drone's position and velocity between detections and corrects its estimate when new measurements are available. For the single-drone case, a standard linear Kalman filter is sufficient, although extended or unscented variants could be applied for more complex non-linear motion models. One downside is the assumption that the posterior be represented as a normal distribution, unlike particle filters such as condensation which can represent multi-model distributions. Alternative deep learning-based methods like DeepSORT, were considered but add complexity unnecessary for a single-object scenario with live tracking. In practice, the SORT (Simple Online and Realtime Tracking) algorithm is used which combines the Kalman filter and the Hungarian algorithm which optimally assigns detections to existing tracks.

3. Failure Modes & Robustness

Real-world conditions pose several challenges that must be addressed for reliable operation. Occlusion is inevitable as the drone passes behind obstacles or temporarily leaves a camera's field of view. Additionally, the drone may only be in a single camera's view, making triangulation impossible. In these instances, the system relies on the tracking algorithm. The Kalman filter mitigates this lack of information by maintaining an estimate until detections resume.

While the RGB pipeline works well in daytime conditions, it may struggle in low-light or nighttime situations. The system can be augmented with thermal or infrared cameras which detect heat from the motor in cooler environments. At long range, the drone may occupy only a handful of pixels, which makes detection difficult. A hybrid setup combining wide-angle and zoom cameras would help. Wide-angle cameras provide broad coverage, while gimbaled zoomed-in cameras offer detailed observations once the drone has been located. Lastly, motion blur can reduce detection accuracy, but appropriate camera settings like high shutter speeds can help.

In terms of latency, this pipeline must work in real time on edge devices. My proof of concept works by analyzing the frames of a prerecorded video. In the real world, it would likely be necessary to reduce the resolution or only do inference on N frames. Also, cameras often have different frame rates, which must be mitigated to ensure synchronization or triangulation will fail. If YOLO has been fine-tuned on the drone dataset, it should be robust to other flying objects such as birds, but would likely fail with multiple drones. In this case, DeepSORT or StrongSORT would perhaps be worth the extra computational complexity to handle target crossings and cluttered backgrounds.

4. Further Improvements

The system described above relies solely on visual information, but additional sensing modalities could greatly improve robustness. As mentioned, thermal infrared cameras can complement RGB cameras by providing contrast between drones and the background in low-light conditions. Acoustic sensors could detect the characteristic sound of drone rotors, allowing localization even when the drone is not visible. Radar offers long-range detection unaffected by weather or lighting and could provide an initial cue for the visual system to direct its focus. Sensor fusion can be implemented using a Kalman filter with multiple measurement models or through probabilistic Bayesian approaches, combining cues from different modalities to produce more reliable state estimates.

5. Proof of Concept

For the prototype implementation, I use the ETH multi-view drone dataset which includes video footage from a ground visual system as well as ground truth 3D trajectories of drones recorded by an RTK system. https://github.com/CenekAlbl/drone-tracking-datasets?tab=readme-ov-file

The code can be found at this github repo: https://github.com/nmorrison01/drone-tracker

Initial results for tracking and detection in 2d can be found in results.pdf