Motivation

- Lab is developing solar powered aircraft
- Maximize efficiency by minimizing energy use
- Remove aircraft landing gear → less weight → less drag → less energy consumed
- Catch aircraft with 15’x15’ net

Proof of Concept Experiment

- 1/5-scale model of net frame in visually noisy environment
- Nvidia Jetson TX1 with Logitech C920 HD webcam
- OpenCV version 2.4.13

Overview

- Use computer vision to detect target frame with OpenCV
- Target frame will hold the net and be bright orange, making it easily distinguishable from the background
- Will be integrated into the UAV flight computer, providing the autopilot the relative aircraft location from the net for landing

Technique

- Calibrate camera and undistort image
- Scale up the image for better accuracy (960 x 720)
- Filter image for orange frame
- Detect contours with cv::FindContours()
- On the contour image, apply cv::HoughLinesP() to contours
- Detect contours on this new image
- Detect corners using cv::ApproxPolyDP
- Prune redundant points from cv::ApproxPolyDP
- Sort points and create lines
- Calculate distance to the sides using:
  \[ \text{Distance} = \frac{\text{actual Width} \times \text{Focal Length}}{\text{Image Width}} \]
- From the distances, can calculate distance in x,y,z directions
- Average the last 3 values, and discard outliers

Results & Conclusions

- Estimation of distance in Z direction is very accurate
- X, Y distances are greatly affected by lens distortion as small changes in detected corners generate large angle differences
- Distortion can be mitigated by keeping the target in the center, or providing some scale based on distance from center of the image
- OpenCV requires lots of hard coding and making the image fit specific guidelines, a learning model could be better.

Future Work

- Process video of the real landing target to verify code
- Create interface with flight control board
- Deploy onto aircraft