Circular Bullseye Detection with the PRG Husky

Team `sudo rm -rf *`

Abhishek Shastry
Department of Aerospace Engineering
University of Maryland
College Park 20742
Email: shastry@umd.edu

Animesh Shastry
Department of Aerospace Engineering
University of Maryland
College Park 20742
Email: animeshs@umd.edu

Nicholas Rehm
Department of Aerospace Engineering
University of Maryland
College Park 20742
Email: nrehm@umd.edu

***Abstract*—In this report, a bullseye target detection algorithm and trajectory planner is implemented on the PRG Husky for ENAE788M: Hands on Autonomous Aerial Robotics. A downward facing Duo camera is used to provide visual feedback of the floor containing a black and white bullseye target. Vehicle location is calculated with respect to the bullseye so that a trajectory through it may be generated and carried out by an odemetry feedback based outer loop control scheme. An EKF is used to refine the vehicle location with respect to the target and to ensure accurate position throughout the flight. Results show that this method is effective in successfully landing on the bullseye target.**

I. INTRODUCTION

Targeted landings with a quadrotor require the use of computer vision to locate the landing site with further computation to determine camera pose with respect to said landing site. A grayscale image from a downward facing camera is enough to locate a distinct bullseye pattern. A Hough circle transform may be used to to find the coencentric circles of the landing site to determine its position within the camera frame. A camera and vehicle pose can be computed from this if the size of the target is known. From there, a simple way-point based controller can be used to safely guide the quadrotor over the target for a precise landing.
Link to the result videos: Click Here

II. TAG DETECTION

Tag detection is achieved using a simple binary thresholding of the grayscale Duo camera image. The binary mask is eroded and dilated slightly to remove any noise. Live auto-exposure was investigated but determined to be problematic in the amount of noise introduced when the target was not in frame. In the future, manual auto-exposure before takeoff will help in maximizing performance across all light levels.

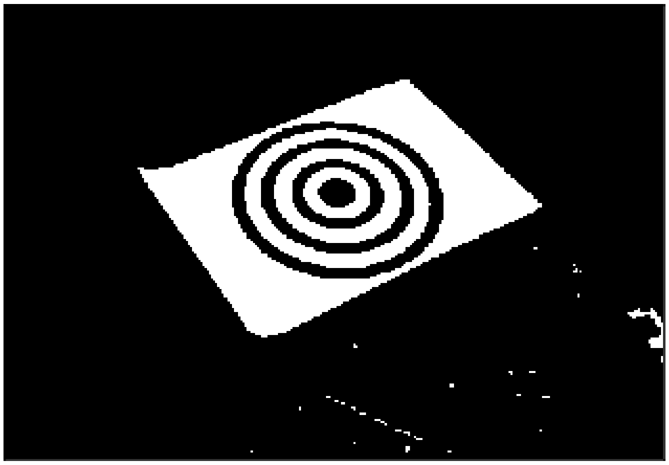

Fig. 1. Binary mask after basic grayscale thresholding

III. ELLIPSE FITTING

After thresholding, circles are fit to the binary image using Hough circles implemented in OpenCV. The centroid of these circles is computed, giving the center of the bullseye target and the center radius. This method works quite well but is only a valid strategy for the downward facing camera where small angles of the vehicle are assumed.

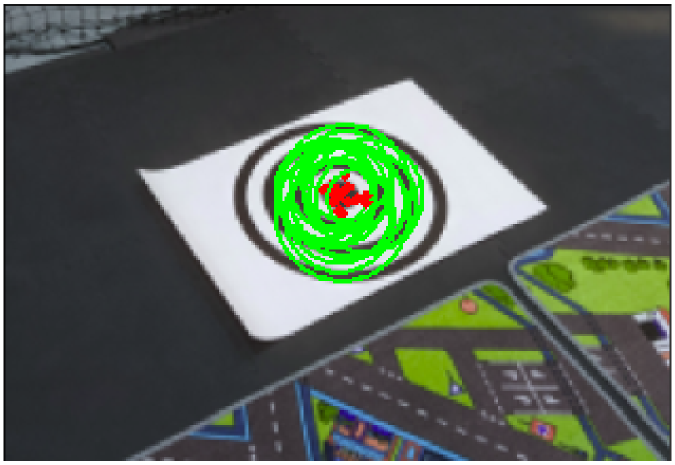

Fig. 2. Circles fitted to binary image, overlaid on original (non-Duo) image as an example. Note: Unrealistic camera orientation while in flight

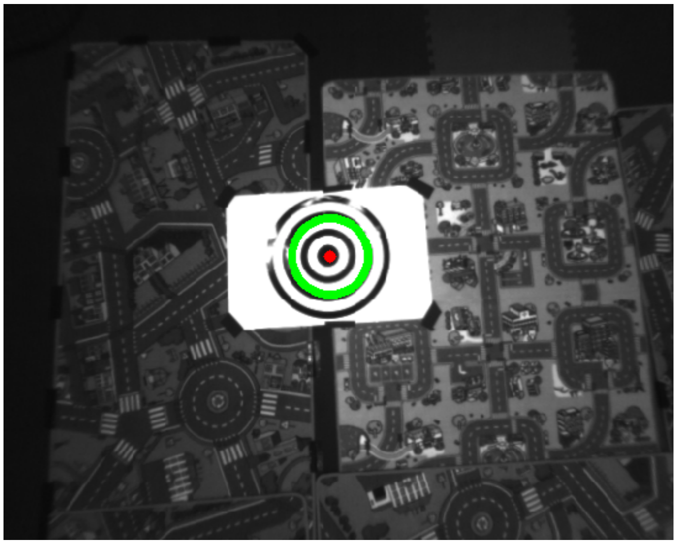

Fig. 3. Example of final circle fitted to bullseye target on Duo image while in flight

IV. CAMERA POSITION RECOVERY

1) Approximate position estimation: Given 2 image coordinates and a real-world size information 5 equations can be constructed. The number of variables to be solved here is 6 (the 2 sets of positions of the 2 points in camera frame). This is represented below.

$$u_1 = f_x \frac{x_1}{z_1} + c_x \tag{1}$$

$$v_1 = f_y \frac{y_1}{z_1} + c_y \tag{2}$$

$$u_2 = f_x \frac{x_2}{z_2} + c_x \tag{3}$$

$$v_2 = f_x \frac{x_2}{z_2} + c_x \tag{4}$$

$$R^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2 \tag{5}$$

Here, as the number of variables to be solved is greated than the number of equations available to us, a small engineering approximation on the values of z_1 and z_2 can be done by assuming that their values are equal. This assumption can also be made true by carefully choosing the two points in the image plane (Ex: The circle's center and far right/left point of the circle). The number of variables is now 5 and a closed form solution for the translation in camera frame is given below.

For $i = 1, 2$

$$x_i = \frac{-\rho f_y (c_x - u_i)}{\sqrt{f_x^2(v_1^2 - v_2^2) + f_y^2(u_1^2 - u_2^2)}} \tag{6}$$

$$y_i = \frac{-\rho f_x (c_y - v_i)}{\sqrt{f_x^2(v_1^2 - v_2^2) + f_y^2(u_1^2 - u_2^2)}} \tag{7}$$

$$z_i = \frac{\rho f_x f_y}{\sqrt{f_x^2(v_1^2 - v_2^2) + f_y^2(u_1^2 - u_2^2)}} \tag{8}$$

$$\tag{9}$$

V. TRAJECTORY PLANNER AND CONTROLLER

A. EKF

An EKF was implemented on top of the estimate of the target pose from vision after converting it to the inertial frame. The inertial frame was chosen because both the state and observation model reduces to identity matrices in this frame and hence EKF reduces to simple linear KF and is therefor an optimal observer which is easier to implement.

B. Search

A search algorithm was implemented to search for the target in the arena after take-off. Search is carried out using a expanding helix pattern bounded by max and min values in Z, such that the tangential speed on the trajectory is a constant. Search will terminate as soon as EKF publishes a bullseye target pose with low enough covariance.

C. Waypoint

The initial waypoint after a lock on the bullseye is confirmed is .75m above the center of the target to avoid ground effect. After an acceptance radius and velocity is met, the vehicle executes the general landing command.

VI. RESULTS AND CONCLUSION

The bullseye target detection and navigation method implemented in this paper was successful in consistently and quickly guiding the PRG Husky to a safe landing. Shortly before the testing time, an issue with the vehicle presented itself in the form of inaccurate position tracking, particularly in the y direction. This was believed to be the result of the onboard odometry malfunctioning, but the cause has yet to be definitively determined. The result of this was bad drift during the search part of the mission. To combat this issue, control gains were tuned down to effectively negate the influence of odometry feedback on the control of the vehicle, leaving position feedback to the locked position of the target from the EKF. Despite this setback, the Husky was able to successfully land numerous times in the allotted 15 minutes and completed the mission in as little as 10 seconds from takeoff. This proves the robustness of the detection and navigation strategy implemented in this paper.