Autonomous Quadrotor Landing on a Moving Platform in Outdoor Environments

Chris Choi*, Stanley Brown† and Steven L. Waslander‡

***Abstract*—**

I. Introduction

II. Related Work

III. Contribution

- Robust Landing
- AprilTag estimation with Kalman Filter
- AprilTag windowing to speed up detection
- Light Invariant AprilTag detection
- Tracking Controller
- Trajectory Planning

IV. Problem Formulation

V. Target Tracking

A popular visual fiducial system called AprilTag [1] was chosen, the system allows us to identify the landing target along with the full 6 dof estimation. There exists multiple implementations of AprilTag, [2] was chosen because it is the most used implementation. In the following we describe steps taken to optimize the AprilTag detection in-order to run the processing at a higher rate.

A. Adaptive Image Processing

The standard implementation of the AprilTag library when processing images of 640 by 480 pixels results with an update rate of approximately 3 to 5 fps on the on-board computer too low for quadrotor controls. To improve performance we adapted the image size depending on the estimated distance between camera and AprilTag.

B. AprilTag Windowing

In [3] an attempt was made to optimize the AprilTag library by reducing the brightness of the image such that the majority of the image is black, we went one step further and masked the lastest image using the last detected AprilTag camera coordinates, rendering everything other than the AprilTag black, with this modification we assumed the image coordinates between frames did not vary too wildly, else the mask is removed and the full image is inputed to the AprilTag detector for processing.

* M.A.Sc. Candidate, Mechanical and Mechatronics Engineering, University of Waterloo; c33choi@uwaterloo.ca

2 M.A.Sc. Candidate, Mechanical and Mechatronics Engineering, University of Waterloo; s52brown@uwaterloo.ca

† Assistant Professor, Mechanical and Mechatronics Engineering, University of Waterloo; stevenw@uwaterloo.ca

Once the center of the AprilTag in camera frame is found the top left and bottom right corners of the AprilTag can be calculated with:

$$X_{\text{top left}} = X - (l/2) - X_{\text{padding}} \tag{1}$$

$$Y_{\text{top left}} = Y - (l/2) - Y_{\text{padding}} \tag{2}$$

$$X_{\text{bottom right}} = X + (l/2) + X_{\text{padding}} \tag{3}$$

$$Y_{\text{bottom right}} = Y + (l/2) + Y_{\text{padding}} \tag{4}$$

Where X, Y are positions of the AprilTag in the camera frame, $X_{\text{padding}}, Y_{\text{padding}}$ are the mask padding in the camera frame, l is the AprilTag side length in meters, and finally $X_{\text{top left}}, Y_{\text{top left}}, X_{\text{bottom right}}, Y_{\text{bottom right}}$ represent the top left and bottom right corners in the camera frame. Adding the mask padding at this point allows the padding to be defined in the camera frame which increases and decreases proportionally in the image frame depending on the depth-distance between camera and AprilTag.

Using the pin-hole camera model, we can convert the AprilTag's corners from camera frame into image frame with the following:

$$x = \frac{f_x * X}{Z} \tag{5}$$

$$y = \frac{f_y * Y}{Z} \tag{6}$$

Where f_x and f_y are the focal length in pixels in the x and y axis, and finally x, y are image pixel coordinates. Using the image coordinates of the top left and bottom right corners of the AprilTag a mask can be created.

C. Nested AprilTag

To address visibility issues of the AprilTag as the field of view of is reduced during landing, we found a unique combination of AprilTag ids that allows us to place a smaller AprilTag in the center of a larger AprilTag (see Fig **??**). By shear coincidence the AprilTag implementation [2] always prefers the smaller secondary tag when both are detected.

D. Illumination Invariant AprilTag

A common issue in computer vision is illumination changes in the environment which in turn interferes with the performance of detection algorithms. From our experience the standard black and white AprilTag fails to be detected if a shadow is casted partially or fully upon the tag features, this is because the AprilTag detection relies heavily on the

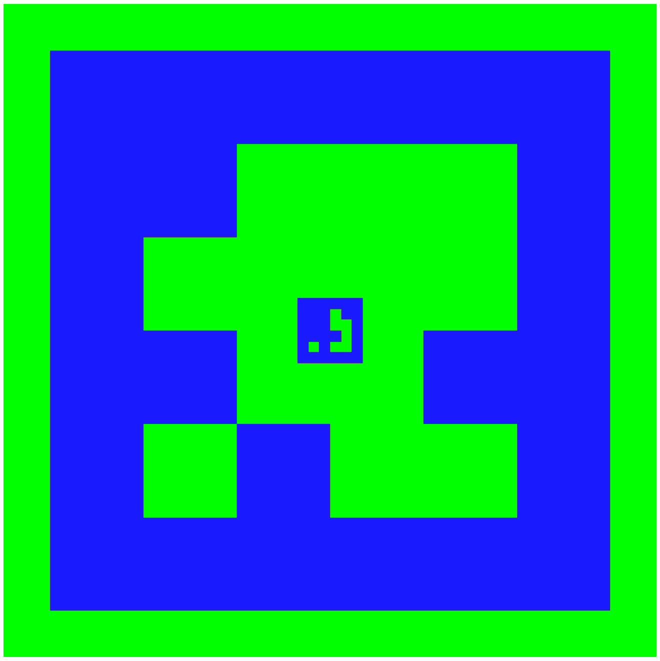

Fig. 1: Illumination Invariant AprilTag

edges and lines of the tag to be able to estimate the 6dof pose.

Reliable image processing regardless of the time of day is vital to the success of target detection. Using methods described in [**?**] we have idenitified the best colours for an illumination invariant AprilTag to be blue and green (see Fig **??**), additionally

$$I = \log(R_2) - \alpha \log(R_1) - (1 - \alpha) \log(R_3) \tag{7}$$

Where R_1, R_2, R_3 are sensor responses (or image channels) corresponding to peak sensitivities at ordered wavelengths $\lambda_1 < \lambda_2 < \lambda_3$.

VI. Target Estimation

The inertial target position, linear velocity, angular velocity and heading are estimated with an Extended Kalman Filter running at 200Hz, the estimation is in turn used for tracking and landing controllers on board the quadrotor.

A. Process Model

We chose to use a two-wheel robot motion model to approximate the forward kinematics of the target, which is given as:

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \end{bmatrix} \\ &= \begin{bmatrix} x & y & z & \theta & v & \omega & v_z \end{bmatrix} \end{aligned} \tag{8}$$

$$\mathbf{u}(t) = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} v \\ \omega \\ v_z \end{bmatrix} = \begin{bmatrix} \mathcal{N}_v \\ \mathcal{N}_\omega \\ \mathcal{N}_{v_z} \end{bmatrix} \tag{9}$$

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \\ \dot{x_5} \\ \dot{x_6} \\ \dot{x_7} \end{bmatrix} = \begin{bmatrix} x_5 \cos(x_4) \\ x_5 \sin(x_4) \\ x_7 \\ x_6 \\ u_1 \\ u_2 \\ u_3 \end{bmatrix} \tag{10}$$

Where the estimator estimates the inertial position in x, y and z direction, heading θ, wheel velocity v, steering angular velocity ω and linear velocity v_z in the z direction. It is worth noting the linear velocity in the x and y direction can be calculated from $v\cos(\theta)$ and $v\sin(\theta)$ respectively. The inputs to the process model are driven by Gaussian White Noise, since we do not possess input information to the two wheel robot motion model.

B. Measurement Model

$$h(t) = \begin{bmatrix} p_{1\times 3} & 0_{1\times 4} \end{bmatrix} \tag{11}$$

VII. Gimbal Transforms

Transform detected target from image frame to gimbal joint frame

$$x_{\text{gjf}} = R_{\text{i}}^{\text{gjf}} R_{\text{cf}}^{\text{nwu}} x_{\text{i}} + t_c \tag{12}$$

Gimbal joint frame to body planar frame

$$x_{\text{bpf}} = R_{\text{gjf}}^{\text{bpf}} x_{\text{gjf}} \tag{13}$$

Body planar frame to inertial frame

$$x_{\text{if}} = R_{\text{nwu}}^{\text{enu}} R_{\text{bpf}}^{\text{if}} x_{\text{bpf}} \tag{14}$$

VIII. Trajectory Planning

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T \\ &= \begin{bmatrix} x & \dot{x} & z & \dot{z} \end{bmatrix}^T \in \mathbb{R}^4 \end{aligned} \tag{15}$$

$$\begin{aligned} \mathbf{x}_0 &= \begin{bmatrix} x_0 & \dot{x_d} & z_d & \dot{z_d} \end{bmatrix}^T \\ \mathbf{x}_d &= \begin{bmatrix} x_d & \dot{x_d} & z_d & \dot{z_d} \end{bmatrix}^T \end{aligned} \tag{16}$$

$$\mathbf{u}(t) = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} a_z \\ \theta \end{bmatrix} \in \mathbb{R}^2 \tag{17}$$

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \end{bmatrix} = f(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} x_2 \\ u_1 \sin(u_2) \\ x_4 \\ u_1 \cos(u_2) - g \\ u_2 \end{bmatrix} \tag{18}$$

$$\begin{aligned} \text{minimize} \quad & J(\mathbf{x}, \mathbf{u}) && (19) \\ \text{s.t.} \quad & \dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) && (20) \\ & \mathbf{x}(t = 0) = \mathbf{x}_0 && (21) \\ & \mathbf{x}(t = t_f) = \mathbf{x}_d && (22) \\ & \mathbf{u} \in \mathbf{U} \quad \forall t \in [0, t_f] && (23) \end{aligned}$$

IX. Quadrotor Control

For target tracking a Proportional-Integral-Derivative (PID) controller was implemented. The estimated target position in inertial frame is converted to body frame and used as control errors to the tracker control

X. Experiment Results

XI. Conclusions and Future Work

References

[1] E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 3400–3407, IEEE, 2011.

[2] "AprilTags C++ Library." `https://people.csail.mit.edu/kaess/apriltags/`. Accessed: 2016-01-12.

[3] K. Ling, U. of Waterloo. Department of Mechanical, and M. Engineering, *Precision Landing of a Quadrotor UAV on a Moving Target Using Low-cost Sensors*. University of Waterloo, 2014.