



A Fuzzy-Logic-Based Solution to Dynamic Target Interception and Landing with a Small Multirotor Aircraft

Nicklas Stockton,* Manish Kumar,[†] and Kelly Cohen[‡]

College of Engineering and Applied Science, University of Cincinnati, Cincinnati, Ohio

The purpose of this work is to demonstrate the efficacy of using a fuzzy logic based approach to successfully perform the landing maneuver of a small multirotor aircraft on a dynamic target at low speeds. The corrective control of the vehicle is exerted based on visual input data from an on-board camera system. Given the constraint of the vehicle size, the computational complexity of image processing and control correction must be low in order to be performed on an on-board computer system-on-a-chip. The design of the controller around visual sensing reduces the dependency of special sensors on-board either the vehicle or target for rendezvous. The use of fuzzy logic enables the controller to adapt to different plant behaviors dynamically without the need for gains scheduling. Fuzzy systems can approximate highly nonlinear functions of arbitrary complexity; fuzzy control, therefore, is a apt fit for highly nonlinear systems which are difficult to control using linear techniques. In addition to being adaptable to nonlinearities, fuzzy logic requires only very basic mathematical operations, making it ideal for platforms which either have limited computational capabilities or need to respond quickly to dynamic conditions. The research is being performed using Robot Operating System to integrate sensing with control motor actuation. All control is performed with Python using OpenCV for image processing. This paper details the simulation setup and image processing algorithms; it also concerns the design and tuning of a simple PID controller. The work is based in high-fidelity simulation in preparation for physical testing on hardware.

Nomenclature

I_{xy}	Pixel intensity value
Δv_x	Control input change to velocity in x -axis
Δv_y	Control input change to velocity in y -axis
Δv_z	Control input change to velocity in z -axis
$\dot{\psi}$	Control input to yaw rate (rotation about z -axis)
d_x	Horizontal offset error from vehicle to target in the body-fixed x -axis
d_y	Horizontal offset error from vehicle to target in the body-fixed y -axis
d_z	Vertical offset error from vehicle to the target
f	Focal length of image sensor
m	Pixel scaling factor
m_{ij}	Raw image moment

I. Introduction

With the ever-increasing proliferation of small unmanned air vehicles (sUAVs) and their use in commercial and emergency response applications, there is a growing need for intelligent, reliable control methodologies

*Senior, Department of Aerospace and Engineering Mechanics

[†]Associate Professor, Department of Mechanical Engineering and Materials Engineering

[‡]Professor, Department of Aerospace and Engineering Mechanics

to safely manage their navigation, especially in possibly congested areas such as disaster areas or urban centers. Commercial delivery companies are moving towards an automated model with reduced human operator intervention to increase the efficiency of their deliveries. One such model consists of a vehicle-based sUAV which departs from the delivery vehicle to make a delivery to a remote residence. Upon completing the delivery, the sUAV returns to the vehicle and docks to receive additional packages. Considering the small target size of a landing platform affixed to the vehicle, and the highly dynamic conditions in which deliveries may be accomplished, the control effort must be accurate and robust in the face of disturbances.

Fuzzy logic provides a way to perform flight control of an sUAV using linguistic variables and control rules, which can be easily understood and tuned using intuition.¹ The resulting control is able to accommodate nonlinearities in the dynamic system such as are found in the situation of an air vehicle to ground transport rendezvous.² Fuzzy logic is also commonly used where there is great need to deal with imprecision or uncertainty,^{3,4} which is inherent in interfacing many different sensors and interacting with real-world situations. Fuzzy systems are highly robust to changes in the plant and environmental disturbances.⁵ Current approaches for developing trajectory paths have focused on time-optimality^{6,7} and not necessarily on lightweight, on-board controllers. In contrast, this work will optimize for reduced control effort and computational simplicity and efficiency. The control developed in this will be largely agnostic to the large linearities inherent in the system dynamics and will apply linguistic variables and logic to intuitively attain the interception and landing. Accuracy will be paramount as the target platform is nearly equally-sized to the sUAV.

II. System Architecture

The research setup involves a quadrotor aircraft of size 450 mm on the diagonal and a mobile rover robot with a 255 mm radius landing platform affixed to it (as shown in figure 1). The quadrotor is controlled by a Pixhawk flight controller which uses the PX4 flight control firmware. This flight controller allows for an on-board computer to take over control of the aircraft via a serial wire connection. A small Linux-based computer is placed on the quadrotor which sends velocity setpoints to the flight controller. All control logic is written in Python using a collection of softwares called Robot Operating System (ROS). ROS allows for easy integration of sensors and control actuation due to a distributed computation framework. As a highly event-driven, publish-subscribe model, ROS maintains an accurate, up-to-date view of internal states which are then exposed to any connected nodes.

An assumption is made that GPS (or some other global positioning) data is available for the simulation; however, this positioning has a margin of error which is far too large to be used exclusively for precision landing. For this reason, an on-board, gimbaled camera will be utilized to detect and locate the target. Using the characteristics of the camera focal length and distortion coefficients, an accurate positional error can be obtained for the feedback control loop.



Figure 1. The test sUAV with the mobile target platform.

III. Simulation

Gazebo is a 3D simulation software which uses open source dynamics engines Bullet, Dart, and ODE to model its components. While Gazebo has very high fidelity simulation capabilities for robots (its initial purpose), the complexities of aerodynamics in general, and multicopter physics in particular, can only be modeled with many simplifications. Even with the simplified dynamics, the simulation environment that Gazebo provides is very useful for high-level controller development. Gazebo allows for the simulation of many sensor types and nicely integrates with ROS and the PX4 flight controller firmware. The high fidelity of the simulation will ease the transition from virtual to real flight as all hardware components have simulated counterparts. The simulation provides a real time interface for tuning the controller with visual feedback. This is the process which was used to tune the PID controller and will be used as well for the tuning of the fuzzy controller.

The most important aspect of sensing for the system is the image sensor. A camera sensor is simulated from the underside of the quadrotor to test the efficacy and efficiency of the computer vision algorithms. Care was taken to accurately represent the field of view and pixel noise of the physical camera sensor to be used. In this way, it is hoped that the controllers developed in simulation will be applicable on the physical system.

IV. Controller

Control is exerted on the vehicle by supplying the flight controller with changes in the velocities, Δv_x , Δv_y , and Δv_z , as well as yaw rate, $\dot{\psi}$. The quadrotor is able to land with any arbitrary yaw angle, ψ , but the assumption is made that some reception mechanism may expect the vehicle in a nose-forward orientation. Distance to the platform is estimated from images taken by the camera sensor. Orientation is estimated only in the last phase of the landing sequence using a robust and efficient visual fiducial tag system⁸ (see figure 2).

A. Computer Vision

Much emphasis is put on the sensing algorithms to be computationally efficient to decrease the load on the on-board computer. For this purpose, only a small number of image processes are required to detect and locate the target. As a first pass, the image is brought into the Hue-Saturation-Value (HSV) color space. This has been shown to be a robust space in which do perform color detection and segmentation in uncontrolled and unpredictable lighting conditions.⁹ A simple thresholding is performed on the image to isolate a sufficiently wide band of yellows to match the color of the target and dilate this to a binary blob. From this binary image, the image moments are calculated by

$$m_{ij} = \sum_{x,y} x^i y^j I_{xy} \quad (1)$$

where I_{xy} is the pixel intensity value for each pixel (x, y) (equal to 1 in this case) and $i, j = 0, 1, 2$. From this it can be seen that m_{00} describes the area and $\frac{m_{10}}{m_{00}}$ and $\frac{m_{01}}{m_{00}}$ describe the centroids \bar{x}_p and \bar{y}_p in terms of pixels. The blob is assumed to be circular and hence a diameter is extracted from the pixel area. Using the focal length of the sensor, these image points are then projected onto the ground plane using the known diameter of the landing pad and a vertical offset estimate is obtained as is shown in equation (2).

$$d_z = \frac{d \cdot f}{m \cdot d_p} \quad (2)$$

where f is the focal length of the camera in units of length, d is the known diameter of the landing pad, d_p is the estimated diameter in pixels, and m represents a scaling factor in units of $\frac{\text{pixel}}{\text{mm}}$ (unity in the simulation).

Assuming that the image plane and the ground plane are parallel, the center of the image can be assumed to point directly below the vehicle and the horizontal offsets to the landing pad can then be calculated as

$$d_x = d_z \cdot \frac{m\bar{x}_p}{f} \quad (3)$$

$$d_y = d_z \cdot \frac{m\bar{y}_p}{f} \quad (4)$$

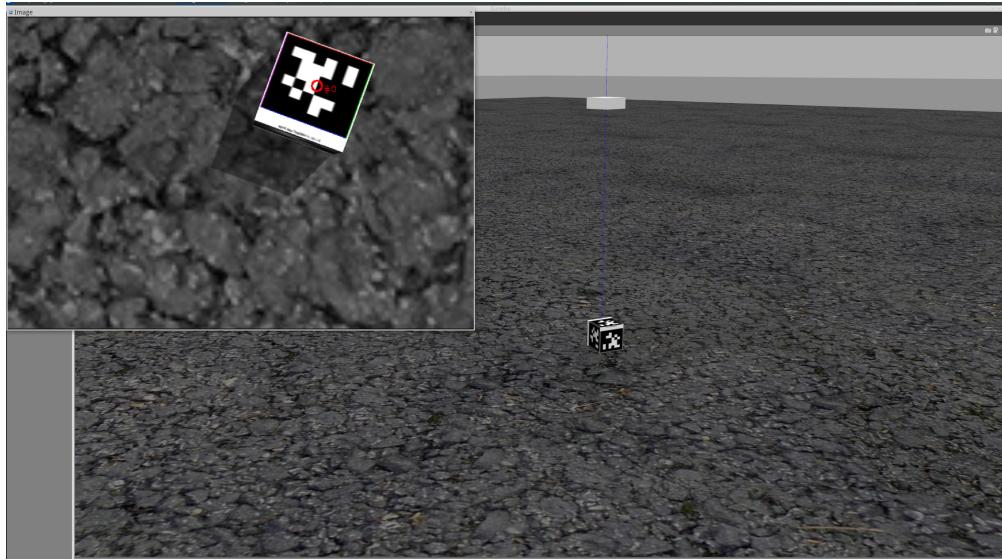


Figure 2. Simulated image sensor detection of AprilTag marker.

As the vehicle approaches the landing pad, the image field of view is overtaken by the landing pad itself and the former segmentation no longer becomes effective. For this purpose, an apriltag⁸ (see figure 2) is placed on the center of the target. Once detection of this tag is achieved, an refined estimate of position and even orientation is obtained with respect to the target. This estimate is used to then orient the vehicle to match the heading of the landing pad (see figure 3).

B. PID Results

A PID controller was created which uses the most recent error estimates and sends velocity requests to the flight controller. Due to the lack of a mathematical model of the system. The PID gains were found by iterating the simulation with various position setpoints and dynamically adjusting the gains while observing the response visually. This closely mimics the method by which PID gains are attained in the tuning of an actual quadrotor by flying a series of test flights and adjusting gain values by feel. In this way, a reasonable response and landing sequence was achieved for first a static target and then repeated for a dynamic target moving with constant velocity. The results are presented in figures 5(a) to 5(b) in which the normed horizontal offset from target is shown in conjunction with the vertical distance to the platform. Viewing the charts, it is apparent that as the vehicle approaches the target, it tends to accumulate horizontal errors and must correct more frequently, particularly in the dynamic case. In both cases, the quadrotor managed to successfully land on the target. In the static case, the final offset from the center of the target was less than 5 cm. For the dynamic case, the error was 7 cm which nearly displaced it from the target surface. In both cases, the sink rate and yaw rates were controlled by simple PD controllers.

C. Fuzzy Results

Though this work is still in progress, promising preliminary results in development of a fuzzy controller have been obtained. Four fuzzy controllers of similar architecture were created as shown in figure 4. Each input fuzzy partition was multiplied by a scaling factor to bring it into the regime of the controller. Likewise, the outputs were then again scaled to match actuation limits. The rule base was developed using common intuition about the system dynamics. A full rule matrix was defined to fully cover the system possibilities. This rule matrix is shown in table 1. The rules are made up of linguistic variables composed into IF-THEN constructions of antecedents and consequents. For example:

IF *error* is N AND *error rate* is Z THEN Δv is SP

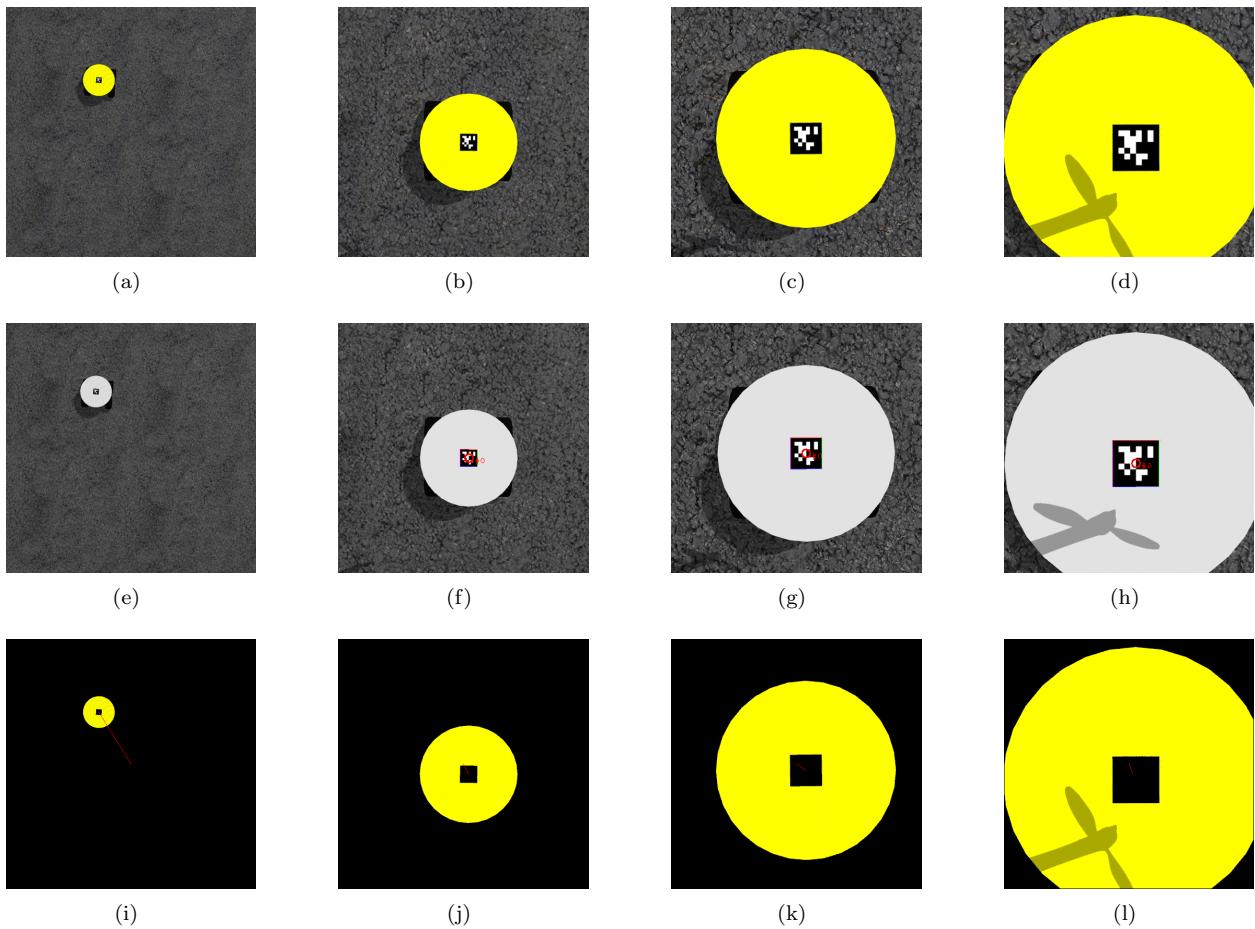


Figure 3. A time series of images taken during the landing maneuver. Note that in figure 3(e), there is no detection of the marker and there remains a slight error in yaw angle as a result. The red line in figures 3(i) to 3(l) represents the horizontal offset of the vehicle to the desired point on the target.

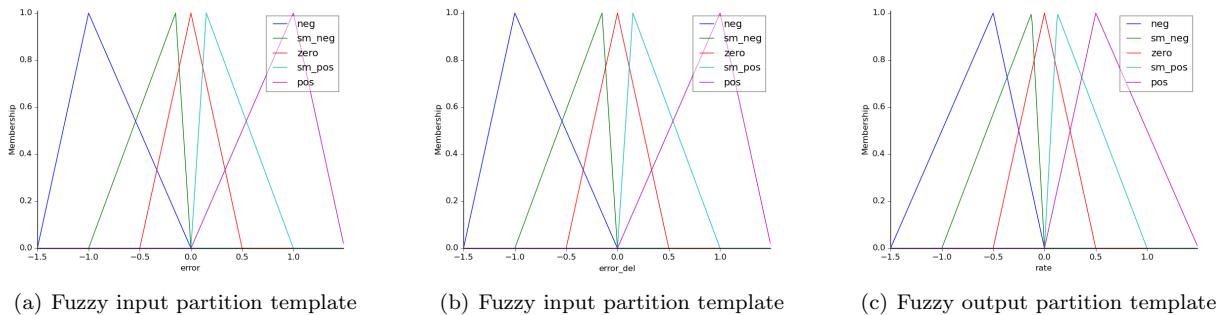


Figure 4. Membership function definitions for fuzzy logic controller.

These rules are intuitive and easy to understand and provide a process by which to lend the controller a decision-making system with a foundation in human reasoning. The tuning process of the fuzzy controller then becomes the task of defining the membership functions which decide how much of each rule should be activated for certain inputs. Triangular membership functions are used exclusively for their simplicity in definition and tuning¹⁰ while the aggregation of rules is the popular min-max method put forth by Mamdani.¹ The membership functions shown in figure 4 are the result of only a small number of iterative tuning steps and were found to be the most effective of the configurations attempted. The fuzzy rule base also provides

ample opportunities for tuning and will have a significant impact on controller performance.

Table 1. Fuzzy rule base

		error				
		N	SN	Z	SP	P
error rate	N	P	P	SP	SP	Z
	SN	P	SP	SP	Z	SN
	Z	SP	SP	Z	SN	SN
	SP	SP	Z	SN	SN	N
	P	Z	SN	SN	N	N

The result of this first-iteration controller was sufficient to land the quadrotor on both the static and constant linear velocity dynamic platforms, though the response to uncertainties is not very stable as of yet. The results shown in figures 6(a) to 6(b) demonstrate the difficulty an untrained controller may have with novel situations. The results are expected to improve dramatically by using genetic algorithms to tune each controller.

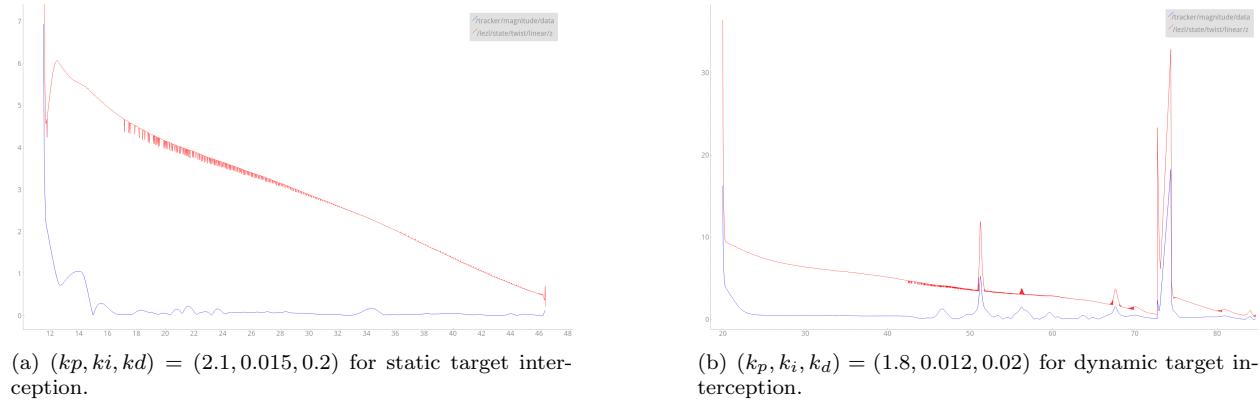


Figure 5. PID controllers for static and dynamic landing

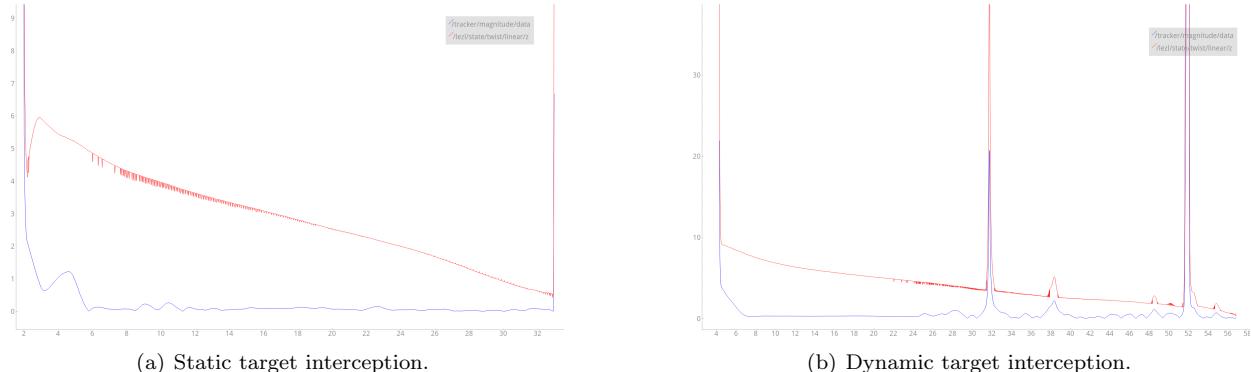


Figure 6. Fuzzy controllers for static and dynamic landing

V. Conclusion

The results presented are preliminary only, but show promising results. The control effort of the untuned fuzzy controller nearly matched that of the tuned PID controller for the static target case. For the dynamic target, while neither controller was tuned to any high degree of accuracy, the fuzzy controller performed at

least as well as the PID controller. It should be noted that the overall positioning control of the vehicle is relatively simple, but as it approaches more closely to the platform, small movements are translated to larger apparent errors and the controller may make bad decisions. This explains the peaks which can be seen in both of the dynamic situations; as the quadrotor becomes very near to the platform, it occasionally loses visual sight of the key target and must then abort the landing and gain altitude to retry.

There is much future work to yet complete. Both the PID and Fuzzy controllers will be tuned by a genetic algorithm which will greatly improve the performance of each]. To prepare the system for real world deployment, the perfect gimbal assumption being made in the simulation will be replaced with a rigidly mounted camera. This will require then applying a rotational transform to each image pixel measurement which will increase the uncertainty in each estimate. Finally, the image processing pipeline must be optimized even further to decrease the computation load on the processor. With the current pipeline, the test hardware processes images at around 2 Hz, which may introduce too much latency in the control feedback. Either the image processing must be sped up, or the delay will need to be handled in some other fashion.

References

- ¹Mamdani, E. and Assilian, S., "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, Vol. 7, No. 1, 1975, pp. 1 – 13.
- ²Ionita, S., "A Fuzzy Approach on Guiding Model for Interception Flight," *Fuzzy Systems Engineering*, Springer Science & Business Media, Jul 2005, pp. 85–111.
- ³*Fuzzy Logic Toolbox 2: User's Guide*, The MathWorks, Inc., 3 Apple Hill Drive, Natick, MA, 01760, 2007.
- ⁴Lee, C.-C., "Fuzzy logic in control systems: fuzzy logic controller. II," *IEEE Transactions on systems, man, and cybernetics*, Vol. 20, No. 2, 1990, pp. 419–435.
- ⁵Janson, A., Stockton, N. O., and Cohen, K., "Genetic Optimization of Fuzzy Logic Control for Coupled Dynamic Systems," *AIAA Infotech @ Aerospace*, American Institute of Aeronautics and Astronautics (AIAA), Jan 2015.
- ⁶Adams, J. C., *An Interpolation Approach to Optimal Trajectory Planning for Helicopter Unmanned Aerial Vehicles*, Master's thesis, Naval Postgraduate School, Monterey, California, 6 2012.
- ⁷Hehn, M. and D'Andrea, R., "Real-time trajectory generation for interception maneuvers with quadrocopters," *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Institute of Electrical & Electronics Engineers (IEEE), oct 2012.
- ⁸Olson, E., "AprilTag: A robust and flexible visual fiducial system," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, May 2011, pp. 3400–3407.
- ⁹Zhao, M., Bu, J., and Chen, C., "Robust background subtraction in HSV color space," *ITCom 2002: The Convergence of Information Technologies and Communications*, International Society for Optics and Photonics, 2002, pp. 325–332.
- ¹⁰Mishra, S., Sarma, I., and Swamy, K., "Performance evaluation of two fuzzy-logic-based homing guidance schemes," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 6, 1994, pp. 1389–1391.