Final Project: Parrot Mambo Minidrone Landing on a Moving Platform

Harshvardhan Sivasubramanian
1233780186
[1]Arizona State University.
[2]MS Robotics and Autonomous Systems.

ABSTRACT

This project builds upon prior experiments by enabling a Parrot Mambo Minidrone to autonomously land on a moving platform. The drone employs real-time RGB color detection to recognize a colored platform mounted on a mobile robot and begins a controlled descent only after successful detection. The system incorporates timing and synchronization strategies to ensure the drone lands before or precisely at the endpoint of the line-following robot's path. This experiment integrates dynamic target tracking, vision-based control, and coordinated interaction between aerial and ground robots, advancing research in aerial-ground robot collaboration.

1. INTRODUCTION.

Coordinated aerial-ground robot systems have broad applications, from warehouse automation to search and rescue operations. This project builds upon the color-based autonomous line-following developed in Lab 3 by integrating a dynamic landing mechanism onto a moving platform.

A Parrot Mambo Minidrone is programmed in MATLAB Simulink to identify a mobile RGB-colored platform mounted on a ground robot. Upon successful detection, the drone performs a precise landing maneuver while the platform remains in motion. This experiment emphasizes real-time vision processing, synchronized movement, and the integration of aerial and ground robotic systems.

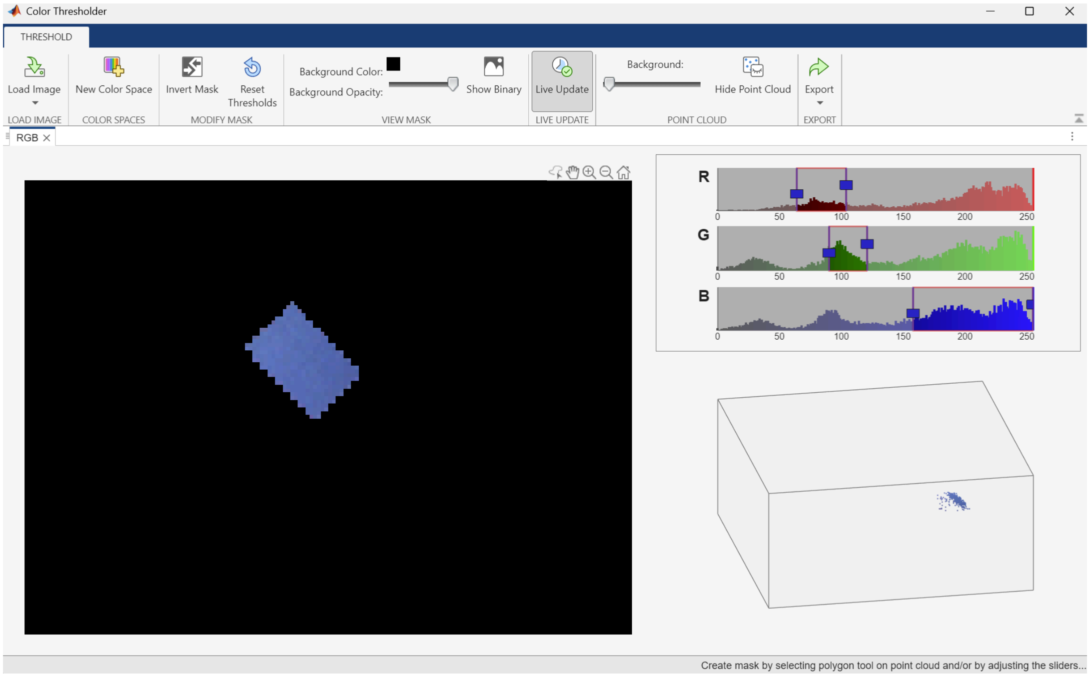
The key challenges addressed in this project include:
- **Real-time target detection** – Identifying and tracking a moving RGB-colored landing platform.
- **Synchronized descent control** – Adjusting the drone's descent in coordination with the velocity of the ground robot.
- **Accurate landing alignment** – Ensuring a smooth landing by preventing overshoot or premature descent.

2. METHODOLOGY.

This system enables the Parrot Mambo Minidrone to autonomously track and land on a moving platform using visual input and logical control based on image segmentation. The process is divided into several core stages:

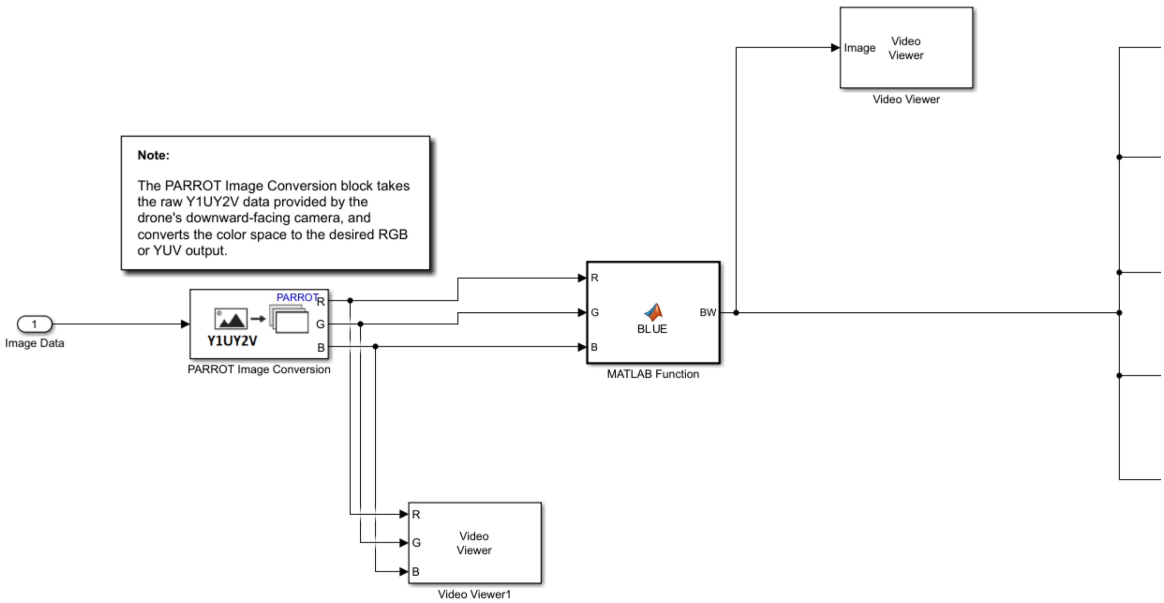
1. Live Video Stream Acquisition
The drone's onboard camera provides a continuous visual stream of the workspace, allowing real-time monitoring of its environment.

2. Defining Color Parameters
MATLAB's Color Threshold Tool is used to set specific RGB values that isolate the desired target color—whether red, green, or blue—corresponding to the platform.



3. Image Analysis in Simulink

Using Simulink's image processing capabilities, the incoming video is analyzed by applying the predefined thresholds to extract the target color regions from each frame.

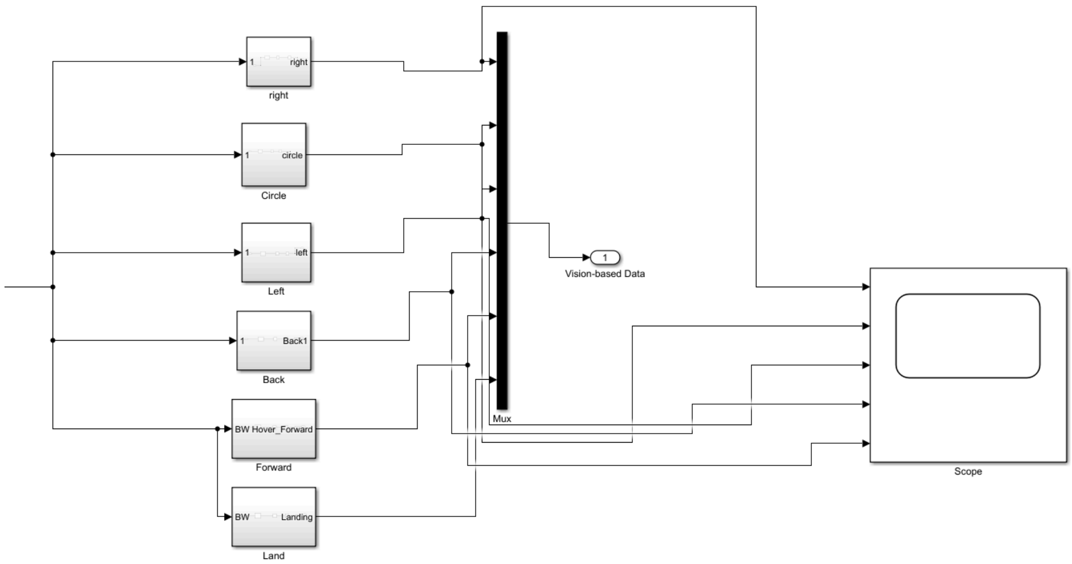



```
function BW =BLUE(R,G,B)
%-----------------------------------------------------
```

```matlab
% Define thresholds for channel 1 based on histogram settings
channel1Min = 64.000;
channel1Max = 103.000;
% Define thresholds for channel 2 based on histogram settings
channel2Min = 90.000;
channel2Max = 121.000;
% Define thresholds for channel 3 based on histogram settings
channel3Min = 157.000;
channel3Max = 195.000;
% Create mask based on chosen histogram thresholds
sliderBW = (R >= channel1Min ) & (R <= channel1Max) & ...
    (G >= channel2Min ) & (G <= channel2Max) & ...
    (B >= channel3Min ) & (B <= channel3Max);
BW = sliderBW;
```

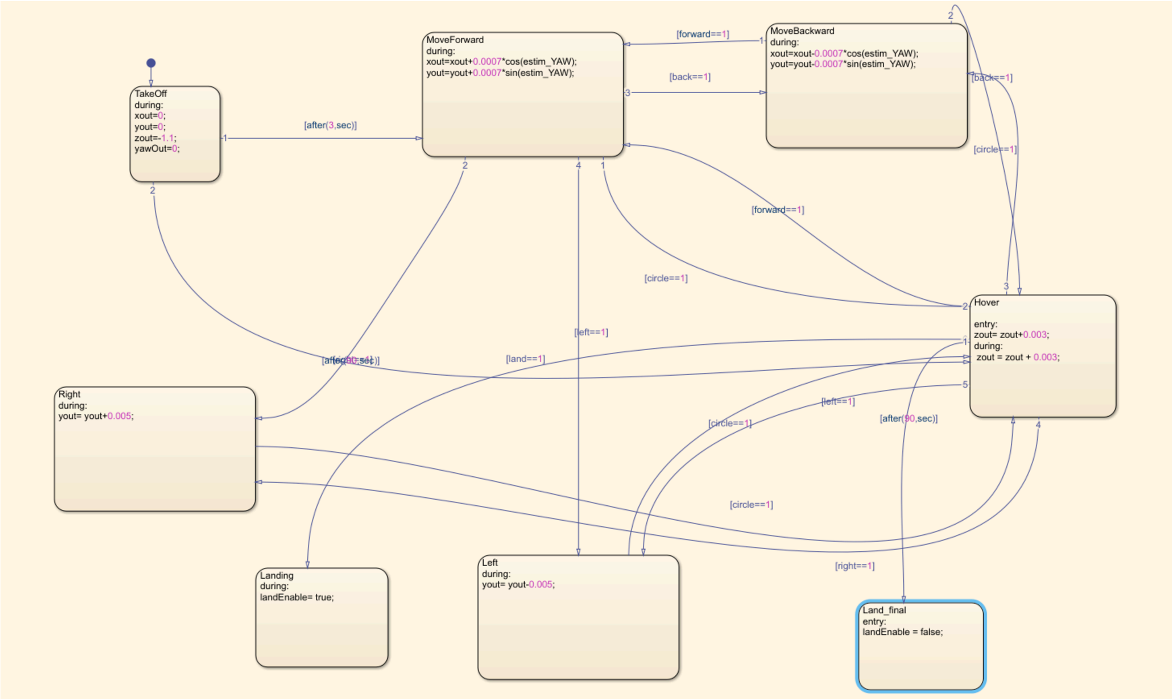
4. Frame Partitioning

To localize the target, the camera view is divided into a 3×3 grid, forming five directional zones: left, right, top, bottom, and center. Each zone guides the drone's navigation decisions.



5. Movement Strategy Based on Zone Detection
The drone responds to the detected location of the platform as follows:

- **Detected at the top** → Move forward
- **Detected at the bottom** → Move backward
- **Detected on the left** → Shift left
- **Detected on the right** → Shift right
- **Detected in the center** → Hover briefly (2 seconds) before initiating landing



6. Descent Execution
A smooth descent begins only if the platform remains in the center zone for a continuous short period. This ensures the drone is accurately aligned before committing to land.

7. Autonomous Navigation Control
The system operates under a control algorithm that guides the drone toward the color-marked mobile platform. It triggers direction changes based on regional logic and ensures stability in hovering before performing the landing maneuver.

3. RESULTS AND DISCUSSION

The system was evaluated in a simulated setup where a mobile ground robot carried a color-coded landing platform. The testing phase focused on the drone's ability to detect, track, and land on the moving target under varying conditions. Several important insights emerged from the trials:

- **Reliable Landing Initiation**
  The drone consistently recognized the RGB-colored platform during trials where the ground robot maintained low to moderate speeds, enabling successful landing sequences.

- **Landing Precision**
  The drone achieved a controlled landing with a typical accuracy range of 10 to 15 centimeters from the center of the moving platform.

- **Motion Prediction and Timing**
  The descent sequence was generally well-timed, as the drone anticipated the robot's movement and initiated landing at suitable moments. However, instances of rapid acceleration by the ground robot occasionally led to imperfect alignment.

- **Sensitivity to Lighting Conditions**
  As encountered in Lab 4, uneven or shifting lighting conditions posed challenges for reliable color detection, affecting the drone's ability to consistently identify the platform.

- **Processing Delays**
  Some minor delays in control response were observed due to the computational load during real-time processing. These latencies were attributed to the onboard processor's limitations, and could potentially be reduced through further optimization of the Simulink model.

- **End-of-Line Landing Challenge**
  When attempting to align the landing with the precise endpoint of the robot's path, accurate estimation of the robot's speed became crucial. This remains a complex aspect of the task and an area for future improvement.

CONCLUSION.

This project validates the potential of implementing vision-guided autonomous landing on a moving platform using MATLAB Simulink. The Parrot Mambo drone effectively identified the RGB-colored target, coordinated its descent with the motion of the ground robot, and executed stable and accurate landings. Looking ahead, enhancements such as fusing IMU and GPS data could improve timing precision, while adopting neural network-based vision systems may offer greater reliability in diverse environmental conditions.

REFERENCES

- MathWorks: Color Detection and Landing - Parrot Example
- MathWorks: Getting Started with Parrot Minidrone Vision
- Lab 3 Report: Vision-Based Line Following for Parrot Mambo Minidrone