Amador Valley High School

(Boreas)

1.Object Detection and Classification:

1.1 System design:

- **Object Detection**: They use the YOLOv5 model running on the drone's onboard Jetson Nano for object detection and localization. This model identifies the ODLCs (Object Detection and Localization Coordinates) and crops out regions of interest, which reduces bandwidth usage and computing cost.
- **Object Classification**: On the ground, the cropped image goes through a Canny edge detector to emphasize edges and remove colors. Then, they use further YOLOv5 models to classify the object's orientation, shape, and character.
- **Background Removal**: The original cropped and resized image is simultaneously processed through two Convolutional Neural Networks (CNNs) to isolate the shape and character. A flood fill technique is used to remove the remaining background pixels. Afterward, the isolated shape and character are color classified using an OpenCV most-prominent-color algorithm.

1.2 Alternatives considered:

In testing, the YOLOv5 ODLC pinpointing model demonstrated significantly better accuracy compared to both EfficientDet-D0 and conventional OpenCV Simple Blob Detection methods, achieving a high mean average precision (mAP) of 0.99 at a threshold of 0.5. The team is currently evaluating an approach called "Slicing Aided Hyper Inference" to further improve accuracy on real-world images.

2.Mapping:

2.1 System design:

Tiling Approach Explanation: In the mapping process, a "tiling" approach is implemented where the map is divided into overlapping rectangles based on altitude and camera specifications. These rectangles are visited in a zigzag pattern, and at each rectangle's center, a photo is taken. If a center falls inside an obstacle or outside the geofence, the camera's gimbal adjusts to capture the image from the nearest accessible point.

Mapping Strategy: The mapping strategy employs a tiling approach, dividing the map into overlapping rectangles. Photos are captured at the center of these rectangles in a zigzag pattern. If a center is obstructed, the camera adjusts to capture the image from the closest accessible point. These images are geotagged, sent to a mapping server, and accurately positioned on a full map using high GPS accuracy and rectangular surveys. This simplified approach delivers high-quality mapping without requiring complex image stitching algorithms.

2.2 Alternatives considered:

The industry standard OpenDroneMap (ODM) framework was initially considered for mapping due to its capabilities in generating geo-referenced images and ease of integration. However, performance tests revealed that ODM was too slow, taking about 10 minutes to process 77 images, which was not suitable for the mission's timeline. As a result, a custom solution was developed and implemented to meet the mission's timing requirements instead of using ODM.

3.Autopilot:

3.1 System design:

3.1.1 Flight Controller and Autopilot:

- The Pixhawk 5X flight controller is chosen for its powerful processor, sensor redundancy, and compliance with Pixhawk standards.
- It is complemented by a Jetson Nano SBC, which handles object detection to reduce bandwidth usage and offload work from the Ground Control Station (GCS).
- QGroundControl is used for mission management and monitoring.

3.1.2 Flight Stack and Control:

- The PX4 open-source autopilot and Drone code flight stack empowers autonomous control with features like waypoint navigation, telemetry streaming, and geofencing.
- An Extended Kalman Filter (EKF) ensures accurate attitude and heading estimation by fusing data from multiple sensors.
- Communication between the autopilot and GCS is facilitated through the MAVLink protocol, supported by QGroundControl GCS and MAVSDK libraries.

3.1.3 Pilot:

- The team's path planner, "Pilot," fetches missions, constructs flight plans, and enables real-time monitoring.
- Flight plans are uploaded to the drone and verified in QGroundControl, enhancing error detection and mission modification capabilities.
- Pilot's Python daemon streams telemetry data to Interop at a 1 Hz rate, providing essential mission oversight.

3.2 Alternatives considered:

3.2.1 Flight Stack:

Criteria	Ardupilot	PX4
Autonomous Flight	Good	Good
SITL simulation	Ok	Good
Community Support	Ok	Good
FOSS Licensing	GPLv3	BSD
Primary GCS	MP	QGC
Standards Compliance	Ok	Good
Aux. actuator control	Good	Ok
Flexibility	Ok	Good

3.2.2 Pilot:

B-spline curves were evaluated to smooth flight trajectory and reduce flight time. However, simulation revealed no significant benefits from this approach.

4.Obstacle Avoidance:

4.1 System design:

Pilot projects the flight area and obstacles onto a 1- meter 2D grid. The A* shortest-path algorithm is applied.

4.2 Alternatives considered:

The A* pathfinding algorithm was evaluated in comparison to artificial potential fields (APFs) and evolutionary path planning models. However, genetic algorithms proved to be slow and vulnerable to getting stuck in locally optimal solutions that were not globally optimal. On the other hand, APFs introduced complexity and often encountered challenges with local minima, offering no substantial performance advantages over the A* algorithm.