Aerial Pathfinding Reconnaissance

Jason Gilman
Max Griffith
Mark Johnson
Dilanka Weerasinghe

FUNCTIONAL SYSTEM REQUIREMENTS

REVISION 1 – Draft
20 September 2020

FUNCTIONAL SYSTEM REQUIREMENTS
FOR
Aerial Pathfinding Reconnaissance

PREPARED BY:

Jason Gilman Date

Max Griffith Date

Mark Johnson Date

Dilanka Weerasinghe Date

APPROVED BY:

Jason Gilman Date

Prof. S. Kalafatis Date

T/A Date

Change Record

Rev.	Date	Originator	Approvals	Description
1	9/15/2020	Jason Gilman		Draft Release

Table of Contents

List of Figures

1. Introduction

1.1. Purpose and Scope

The Aerial Pathfinding Reconnaissance System (APRS) is intended to provide an effective approach to generating navigation instructions for autonomous vehicles. The APRS is able to efficiently analyze a user defined area such that an autonomous vehicle can traverse from endpoints that are input by the user. The APRS is designed so that the functionality of the system is abstracted from the end-user. A front facing UI will allow any person with basic knowledge in setting up drones to launch, update, and suspend the system. Figure 1 below shows a high-level overview of the APRS.

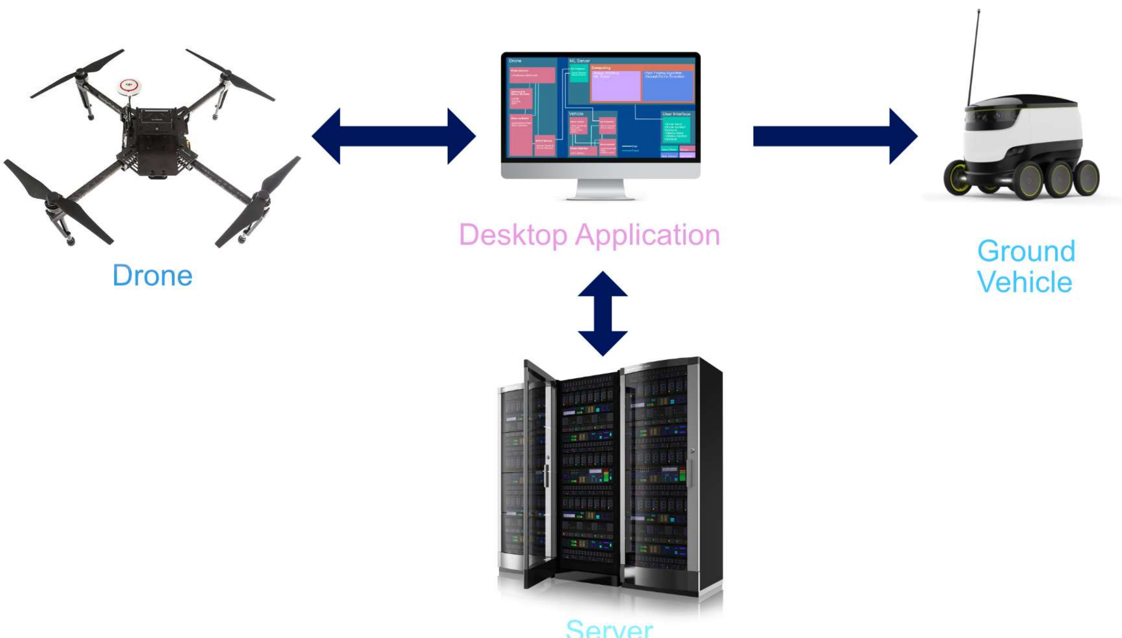

Figure 1. High-Level System Overview

The Aerial Pathfinding Reconnaissance System (APRS) will operate through interaction with a front facing UI. The UI will be run on a local machine that will serve as the main node in the APRS network, which connects all of the devices, vehicles, and software encapsulated in the system. Data collected by the drone will be sent to the local machine so that software can be utilized to generate navigation instructions. These instructions will then be routed to the ground vehicle where they will be used to traverse the user defined area.

The following definitions differentiate between requirements and other statements.

Shall:	This is the only verb used for the binding requirements.
Should/May:	These verbs are used for stating non-mandatory goals.
Will:	This verb is used for stating facts or declaration of purpose.

1.2. Responsibility and Change Authority

Jason Gilman, the team leader, has the responsibility to confirm that the APRS follows all of the defined specifications. Changes made to project documents or system specifications must be approved by Jason Gilman.

2. Applicable and Reference Documents

2.1. Applicable Documents

The following documents, of the exact issue and revision shown, form a part of this specification to the extent specified herein:

Document Number	Revision/Release Date	Document Title
IEEE 802.11	2016	IEEE Standard for Information technology—Telecommunications and information exchange between systems Local and metropolitan area networks—Specific requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications
PEP 8	2001	Style Guide for Python Code
Part 107	2018	Fact Sheet – Small Unmanned Aircraft Regulations (Part 107)

2.2. Reference Documents

The following documents are reference documents utilized in the development of this specification. These documents do not form a part of this specification and are not controlled by their reference herein.

Document Number	Revision/Release Date	Document Title
	2019/04/11	Tarot 650 User Manual
	2018/08/10	DJI MATRICE 100 Product Release Notes
V1.6	2016/03	DJI MATRICE User Manual
V1.0	2015/09	Intelligent Flight Battery Safety Guidelines
MIL-STD-704F	1991/5/1	(D.O.D.) Aircraft Electrical Power Standard
SAE AS1212	2021/12/1	Commercial Electrical, Aircraft Characteristics

2.3. Order of Precedence

In the event of a conflict between the text of this specification and an applicable document cited herein, the text of this specification takes precedence without any exceptions.

All specifications, standards, exhibits, drawings or other documents that are invoked as “applicable” in this specification are incorporated as cited. All documents that are referred to within an applicable report are considered to be for guidance and information only, except ICDs that have their relevant documents considered to be incorporated as cited.

3. Requirements

In the following section, “Aerial Pathfinding Reconnaissance System” or “APRS” will refer exclusively to the entire system. This will include the power distribution, whether that be for the aerial drone or the land vehicle, the camera to capture ground footage and LiDAR to avoid aerial objects that may be in the drones’ path, GPS systems to triangulate the precise location of the drone and to calculate the land vehicle’s location as it is following the path, the DJI drone controller which will be used to accommodate for any needed flight path updates, a ground controller that will act as the bridge from command to the motor of the land vehicle, computer vision and mapping where video data will be consumed in piece to produce stereographic ground views without moving obstructions which will be analyzed to produce a depth estimation map, pathfinding and drone detection which will analyze the vehicles start and end positions in an environment and the map itself to produce a path and creating commands for the land vehicle to follow, and device networking and UI which will route the data through a wireless network and the UI will keep the user informed of the system’s progress.

3.1. System Definition

The Aerial Pathfinding Reconnaissance System (APRS) is composed of four discrete entities:

- **Vehicle:** The ground vehicle will be used as a demonstration of the APRS. It contains no visual sensors for avoiding obstacles. Instead, it will receive directions from the locally run application and execute them accordingly.

- **Drone:** The aerial drone will be used to gather data on the operating environment of the ground vehicle. It will use an onboard transceiver to wirelessly relay video information for further processing on the local machine.

- **Laptop/Local Machine:** The local machine will run a desktop application that will use video information from the drone to produce directions for use by the land vehicle. To do this, it will stitch video frames together to use a map and to construct a stereographic view from which to infer depth. Then, a pathfinding algorithm will be used to generate the final directions. Optionally, these processes may be offloaded to a cloud server to increase computational power.

- **User Interface:** The user interface provided in a desktop application will form the frontend of the system, relaying information about progress and status to the user, and allowing the user to configure pre-flight parameters such as e.g. aerial path, altitude, etc.

Shared among these four elements are four subsystems. Some of these subsystems are spread across multiple entities, while some are limited to one location as shown in **Error! Reference source not found.**

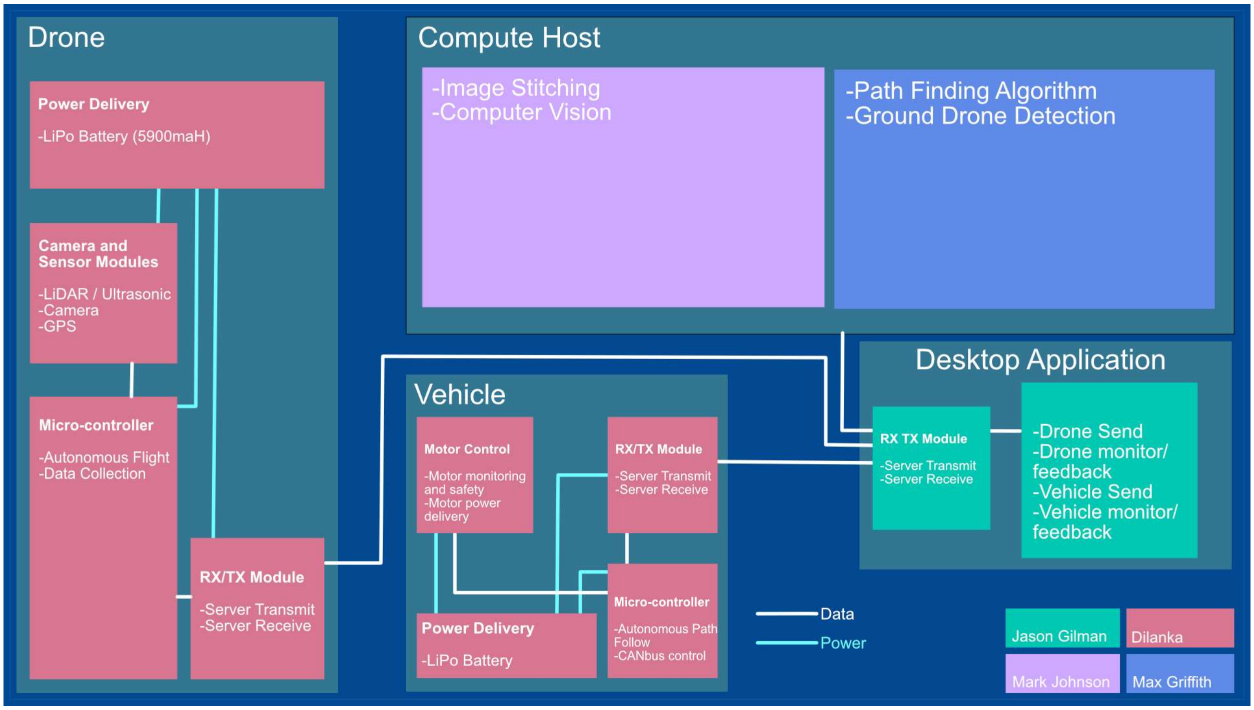

Figure 2. Functional Block Diagram

The color-coded sections of the diagram show which person will be managing their development, with each team member on one subsystem according to the following mapping:

- Jason Gilman: Device Network and User Interface
- Dilanka Weerasinghe: Data Collection and Drone Control
- Max Griffith: Pathfinding and Drone Detection
- Mark Johnson: Computer Vision and Mapping

The diagram also displays points of communication or otherwise shared resources. Where not specified, it is assumed that resources available to one entity are shared by all subsystems within it.
For example, consider how all of these components are used in our system. First, the drone uses its components (power, cameras, etc.) to collect imaging data of the environment. Then, through its connection to the local machine facilitated by their transceiver modules, this data is relayed for processing. Progress on this task is relayed to the user through the interface, which can also be used to set flight parameters and other options.

The processing of this data occurs in the following fashion: First, the imaging data (in the form of a video) is stitched together to form a complete bird's eye view of the environment. After depth estimation and obstacle detection is complete, the pathfinding algorithm and ground drone detection work together to find a path from its current position to the target. Finally, after a path is found and navigation instructions are created, they can be sent to the ground vehicle (which uses its components to carry out these instructions).

3.2. Characteristics

3.2.1. Functional / Performance Requirements

3.2.1.1. Flight Survey Area

The APRS shall have capacity to survey an area of ~1km in one flight.

> *Rationale: As required by the sponsor, the initial surveying flight must cover a defined plot.*

3.2.1.2. Payload Capability

The APRS shall have capacity to carry 1400 grams in addition to its weight.

> *Rationale: In accordance with the specifications of the drone it cannot exceed a total weight of 3680g.*

3.2.1.3. Movement Characteristics

The APRS aerial drone shall have a maximum yaw angle of 200 degrees and a maximum tilt angle of 35 degrees. This drone shall not fly faster than 15m/s.

> *Rationale: A characteristic of the Tarot 650 V2.1 drone details the above figures that the programs will not attempt to exceed.*

3.2.1.4. Flight Range

The APRS shall have capacity to travel 3.2 m away from the user starting location.

> *Rationale: The telemetry modules on commercial drones can only communicate at a functional maximum of this distance.*

3.2.1.5. Flight and Land Duration

The system aerial drone is operational for up to 25 minutes in the air. The land vehicle will be able to follow various paths for up to 30 minutes.

> *Rationale: This aerial drone contains 1 single 5000mah LiPo batteries. Base conditions through Tarot testing show that it can remain in the air stationary for 40 minutes. The Land drone will have a 4000mah battery that will allow it travel long distances.*

3.2.1.6. System Lifetime

The Drone will not be operational after a single flight path is followed. All components of the drone, the flight controller and the land drone will require a charge after operation.

> *Rationale: After each use of the APRS, components of the system require maintenance. The aerial and land drones' batteries must be charged before attempting to use again.*

3.2.1.7. System Networking

When transferring data between devices, the Aerial Pathfinding Reconnaissance System requires that the devices must be close enough such that the devices can be connected to a shared wireless local network.

Rationale: The APRS will utilize wireless local area networks to communicate between devices. In order to facilitate communication between devices on WLAN, the devices must be in relatively close proximity to each other.

3.2.1.8. Obstacle Detection

The APRS shall be able to accurately detect obstacles as tall as 5% of its altitude.

Rationale: As defined by the sensitivity of the sensors onboard the aerial drone, the APRS will be able to detect objects that meet a height requirement.

3.2.1.9. Optional Internet Availability

Optionally, processes may be offloaded to a GPU accelerated cloud server. To utilize this, internet access must be available to the local machine that is running the system's desktop application.

Rationale: In order to communicate with the hosted server, the local network node must have an internet connection.

3.2.2. Physical Characteristics

3.2.2.1. Drone Mass

The mass of the drone in the APRS shall be less than or equal to 3 kilograms.

Rationale: This is a requirement of the drone manufacturer to maximize flight time while maintaining all necessary functions of the drone. The drone body weighs 2180 grams and the all addition sensors weighing less than 800 grams.

3.2.2.2. Land Vehicle Mass

The mass of the land vehicle in the APRS shall be less than or equal to 5 kilograms.

Rationale: This is a requirement posed by our customer detailing that the system should be transferable by hand to any location.

3.2.3. Software Characteristics

3.2.3.1. Programming Style

Software subsystems written in Python shall conform to the PEP 8 Style Guide.

Rationale: The software components of the APRS should be written in a consistent programming style for maintainability purposes.

3.2.3.2. Source Commenting

Source code for software subsystems shall contain at least one comment for every 5 SLOC, and a documentation string at the beginning of each function definition.

Rationale: The software components of the APRS should contain sufficient internal documentation to understand and maintain the source code.

3.2.3.3. Error Handling

Upon any unrecoverable error conditions, software subsystems will use the standard exception handling capabilities of their source language.

Rationale: The software components of the APRS should utilize a consistent error handling mechanism to ensure ease of integration.

3.2.4. Electrical Characteristics

3.2.4.1. Inputs

a. The presence or absence of any combination of the input signals in accordance with ICD specifications applied in any sequence shall not place the user in a state of danger, place the APRS in a state of danger, reduce its life expectancy, or cause any malfunction, either when the unit is powered or when it is not.

b. No sequence of user command shall damage the APRS, reduce its life expectancy, or cause any malfunction.

c. In the event of fatal malfunction, the APRS shall be taken control by a user and may be stopped at any time to prevent such fatal error.

Rationale: By design, should limit the chance of damage or malfunction by user/technician error and by error of acts of god or unpredictable situations.

3.2.4.1.1 Power Consumption

a. The maximum peak power of the aerial system shall not exceed overcome the limits of the battery compartment on the drone. The voltage will not exceed 22.2V and in accordance with a maxima 4.5 A.

Rationale: A requirement drone systems to regulate power and maintain stable flight.

3.2.4.1.2 Input Voltage Level

The input voltage level for the APRS aerial drone shall be less than +29 VDC. At system power on the normal voltage limit may be exceeded at battery start.

Rationale: Aircraft bus specification compatibility, MIL-STD-704F

3.2.4.1.3 Charging and Capacity

The charge capacity of the battery will not exceed 99% of the 22.2 V battery operation. The maximum charging power shall not top 180W. The battery will display a low battery state when under 12.5% capacity.

Rationale: Limitations and safety features of LiPo batteries. By design to maintain battery and drone health.

3.2.4.2. Outputs

3.2.4.2.1 User Interface

The APRS will provide a UI encompassed in the system's desktop application that will update the user on the system's progression. When the system progresses from data collection to software analyzation, and ultimately to preparation of the land vehicle, the user will be visually updated within the UI.

> *Rationale: The APRS has many steps and allowing the user to interact through a simple UI will decrease the learning curve for the system.*

3.2.4.2.2 Low Battery Indication

The APRS will provide both an update to the UI and will enable a physical notification on the drone that it is incapable of traveling.

> *Rationale: To maintain the safety of the drone and the any nearby individuals.*

3.2.5. Environmental Requirements

The APRS shall be designed to withstand and operate in the environments and laboratory tests specified in the following section.

> *Rationale: This is a requirement specified by our customer due to constraints of their system in which the APRS is integrating.*

3.2.5.1. Pressure (Altitude)

The APRS can be operated within 0 to 120 meters above ground level, or 0 to 120 meters above a structure.

> *Rationale: This operating range is in compliance with FAA Small Unmanned Aircraft Regulations (Part 107).*

3.2.5.2. Thermal

The APRS shall be operated within a temperature range of -10° C to 40° C.

> *Rationale: As per the user manual, the battery that is included with the drone can operate within the temperature ranges of -10° C to 40° C.*

3.2.5.3. Inclement Weather Conditions

The APRS shall not be operated in any condition that involves inclement weather such as rain, snow, dust storms, high-speed winds, hail, or thunderstorms.

> *Rationale: The drone will not operate in any condition of inclement weather to ensure the safety of the aerial drone as it is not waterproof and cannot operate in extreme conditions.*

3.2.6. Failure Propagation

3.2.6.1. Aerial Drone Flight Failure

Upon error during flight time, the user should immediately suspend drone operation through use of the remote control provided.

Rationale: Per the requirements to the Texas A&M University ECEN Department, a user must be able to suspend the operation of the aerial drone using the provided remote control upon flight error.

3.2.6.2. Low Battery

In the event of low battery, the drone will suspend operation and return either to the user or to a location where it can be recovered. The drone will not continue operation in a state of damage or irresponsible condition.

Rationale: To maintain the safety of the drone and the any nearby individuals.

4. Support Requirements

4.1. Materials Provided

The following materials will be provided to the user.

4.1.1. Aerial Drone System

A fully encapsulated aerial drone system will be provided to the user. This will include sensors such as camera, LiDAR, and ultrasonic. A microcontroller onboard the drone will fulfill the interaction between hardware and software and provide functionality to receive and send data from the machine running the system's application. The aerial drone system will contain batteries and power distribution such that all electrical components are able to function properly.

4.1.2. Land Drone System

The land drone system will be provided to the user. This system will contain the vehicle's chassis, wheels, and motors. A microcontroller will be attached to the chassis so that software can be implemented to drive the motors and receive data from the local machine running the system's application. The system will contain batteries and power distribution such that all electrical components are able to function properly.

4.1.3. Desktop Application

Software will be provided to the user that will serve as the main interface to the user. The software will provide capabilities such as device networking, data transmission and analysis, and control over the system's processes.

4.2. Local Machine Requirements

The user must possess a local machine such as a laptop that is capable of running the system's desktop application. The recommended specifications are as below:

Processor: Intel Core I5 7th Generation or equivalent
RAM: 8GB DDR3
Operating System: Windows 10
Storage: 5GB

4.3. System Malfunction or Destruction

Upon destruction of the aerial or land drones, a professional will be required to service the affected system. If at some point a portion of the system malfunctions, the user will be updated within the UI, and will be able to make changes such as restarting or suspending the system.

Appendix A: Acronyms and Abbreviations

APRS	Aerial Pathfinding and Reconnaissance System
GPS	Global Positioning System
GUI	Graphical User Interface
ICD	Interface Control Document
SLOC	Source Lines of Code