UAV Austin
AUVSI SUAS 2018 Technical Design Paper

The University of Texas at Austin

Abstract

In the past decade, UAS aircraft have become increasingly prominent in tasks spanning military reconnaissance missions, commercial applications, and even personal use. As the demand for these aircraft continue to rise, the need for innovation in technology, reliability, and safety have also become objectives in the field of UAS development.

The AUVSI SUAS competition serves as a hub for creativity in design and mixing of ideas. Incorporating mission objectives relevant to the cutting-edge technology and capabilities of the UAS field, the competition challenges teams from universities across the world to develop exceptional unmanned systems.

The University of Texas at Austin took a systems engineering approach to maximize performance of the aircraft under given situational constraints while maintaining gracious professionalism and a mindfulness of excellence in the final project. Building upon the 2016-2017 project, UAV Austin was able to effectively carry out the process through the formation of a research and development team, multiple design iterations, and systems and flight testing in order to complete the second iteration of the aircraft. The team is comprised of a diverse group of undergraduate students spanning across various engineering disciplines, with an impressive 50% freshman composition. An emphasis was put on freshman involvement to bridge the knowledge gap and build a strong foundation in moving forwards, including the 2018-19 aircraft, Perspective. The following technical report seeks to exhibit the objectives, mission considerations, aircraft components and design procedures. Building upon the 2016-2017 project, UAV Austin was able to effectively carry out a successive iteration of its current aircraft.

Contents

1 Systems Engineering Approach

INCOSE defines systems engineering as an interdisciplinary area of engineering that synthesizes multiple considerations simultaneously in order to optimize the design process and ensure that the customer and stakeholder's needs are met and exceeded. [3] In the INCOSE process, design steps are iterative and performed in parallel. This non-sequential approach to completing a project allows for the consideration of many factors, creating an effective and multifaceted process of design and production. UAV Austin structured the project in a similar manner as INCOSE's systems engineering model. This was carried out by first stating objectives and investigating alternatives. Through modeling, integrating, launching, assessing, performance testing and re-evaluating the system, the project was executed with attention to the quality and elegance of the final product. The objectives were defined using the previous year's competition rules. Alternative approaches to the systems were integrated into last year's UAS in order to ensure improved flight performance and overall higher capability to carry out missions. Systems were then rendered and assembled into a complete UAS model using SolidWorks. After the completion of the CAD model, all of the systems were integrated into the UAS. The aircraft then underwent extensive performance testing to ensure that the entire system operated properly. Through the assessment and analysis of multiple flight and systems tests, the performance of the aircraft and each individual subsystem was evaluated. These evaluations led to successive iterations of improvements which were then integrated into the UAS. The dynamic INCOSE model greatly streamlined the design process.

1.1 Mission Requirements Analysis

The foundation of any system is governed by the set of constraints that define the system requirements. UAV Austin analyzed the constraints defined by the competition rules, created a weighted table based environmental factors (Table 1), and defined a set of derived requirements for the UAS. The mission, as defined by the AUVSI SUAS competition, requires autonomous flight, obstacle avoidance, object detection, classification, localization, and air delivery.

1.1.1 UAS Derived Requirements Overview

The system is designed with the mission requirements in mind, while also considering the team's previous experience and budget constraints. Other restrictions include a maximum takeoff weight of 55lbs, a maximum airspeed of 70 KIAS, and standard, low risk fuel or batteries. The UAS must do the following:

1. The UAS shall be capable of autonomous flight.
2. The UAS shall accurately fly to each waypoint while remaining inside the flight boundaries.
3. The UAS shall avoid stationary and moving obstacles, whose locations are received from the interoperability system.
4. The UAS shall take images/video of approximately 0.1 square miles in under 25 minutes to identify target characteristics and location.
5. The UAS shall accurately deliver an 8oz water bottle that will rupture on impact within the drop location. [2]

1.1.2 Ground Station Derived Requirements Overview

1. The ground station shall display a map showing flight boundaries, UAS position, UAS speed, altitude and other competition elements for the competition judges.
2. The ground station shall send mission commands to the UAS and receive imagery from the UAS.
3. The team shall have personal protective equipment (PPE) which includes, at minimum, proper tools, gloves, eye protection, and hearing protection when appropriate. Safety risk mitigation shall also be implemented, which includes team training, checklists, and radios for communication. The team shall have equipment (first aid kit and fire extinguisher) to respond quickly to emergencies.
4. The ground station shall receive mission details and submit mission deliverables using the interoperability system.
5. The ground station shall be portable and easy to set up in less than 20 minutes. [2]

1.2 Design Rationale

Due to budgetary and time constrictions, it was decided that the same airframe used from the 2017 competition would be used again for this year's competition. For the current iteration, the tail boom was redesigned with new carbon fiber tubes and a new boom-to-tail interface. The nose cone was redesigned to accommodate a new camera and camera gimbal assembly. The empennage of the UAS was also modified to optimize flight control and simplify pitch and yaw tuning.

The emergence of new production capabilities such as composite layups and advanced additive manufacturing brought many ideas for innovation. These facilities combined with a deepening interest in aircraft design led the UAV team to consider creating a long term fixed wing aircraft set to debut in the 2018-19 season while simultaneously implementing experimental refinements on last year's reliable aircraft, Sir-Vay-Lance. These improvements include a new 3D printed nylon reinforced ABS camera gimbal that is optimized for the new camera, a new fiberglass-epoxy nosecone, and a complete overhaul of UAS avionics from the previous year.

1.2.1 Environmental Factors

UAV Austin began the Fall 2017 semester with a budget of $9,700 for system design and travel expenses to Maryland. $5,000 of the budget was set aside for traveling to the competition this year, and the remaining $4,700 was set aside to develop this year's and next year's competition aircraft. 11 members from the 2016-2017 academic year remain active on the team this year, while the rest of UAV Austin consists of 14 new members. The majority of the team is Aerospace and Computational, while 8 percent are Mechanical and 5 percent are Chemical Engineering. Each member is expected to uphold a certain amount of hours per week, with a minimum of six hours or two meetings per week. To determine the system design approach, the Program Manager, Chief Engineer, and Chief Software Engineer analyzed the team's abilities. The team has experience in CAD modeling, aerodynamic analysis, electronic circuit design, computer vision, computer networks, and full stack development. The analysis of the team's skills motivated the decision to design a plane uniquely suited to the competition's requirements. It

Task (Weight %)	Description	Requirement for Accomplishment
Autonomous Flight (35%)	• Fly the mission without a manual take-over • Fly through each waypoint within a 100 ft margin	• Flight testing of the developed autopilot to ensure performance within the 100 ft error margin
Timeline (20%)	• Complete Flight and post-processing in minimal time • Refrain from taking a time out • Ensure optimal flightline communication and established flight plan	• Optimization of UAS for flight time • Full mission test flights
Object Detection, Classification, Localization (20%)	• Identify target shape,color, character, and orientation • Identify the GPS location of each target Submit targets autonomously during flight • Submit targets via Interoperability System	• Create accurate simulation of competition environment • Optimize pixels on target to meet both user defined constraints and UAS flight capabilities
Air Delivery (10%)	• Deliver the payload (water bottle) to the target within a 10 ft margin of error	• Development of a deployment system that accommodates the payload • Modeling and testing of said system, with precise results
Operational Excellence (10%)	• Maintain strong team communication and developed roles with dedicated checklists	• Practice mission tests and detailed explanations of each member's role
Obstacle Avoidance (5%)	• Avoid stationary obstacles • Attempt to predict and fly around moving obstacles	• Establish strong understanding of flight characteristics • Create mission plan around stationary obstacles and respond quickly to moving obstacles

Table 1: Mission Requirement Analysis

was recognized early that new manufacturing techniques and design features were needed for the new aircraft design. Sir-Vay-Lance was chosen to be the technology demonstrator for these new techniques and design features. In order to test the performance of these new design features, the team held frequent flight tests at the Austin Radio Control Airfield. With the wide range of skills utilized to complete the project, UAV Austin is confident the aircraft can accomplish all tasks at the competition.

1.2.2 Design Selection Criteria

A fixed-wing design was the natural choice of platform due to the large payload requirement of the competition. The team and department faculty having extensive experience in fixed-wing design also prompted the team to take this approach. In order to select the ideal platform for the UAS, weighted criteria were made to construct trades on the design parameters of the aircraft. Moreover, the team concluded that optimization of the 2016-2017 system in tandem with designing a new long-term project for the next two seasons would be the most efficient and successful for UAV Austin.

2 Systems Design

2.1 Aircraft

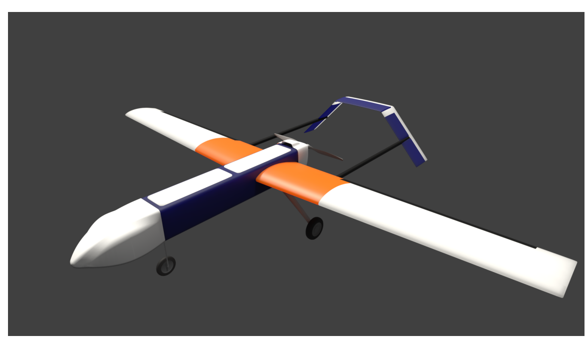


Figure 1: Perspective View of the Aircraft

2.1.1 Airframe

The Hugin II is a pusher configuration aircraft. It has a fuselage length of 42.5 in and an effective wingspan of 102.25 in. The design gives a large amount of space to add components, which allows for the center of mass to be adjusted as needed. Additionally, the large space allows for easy access to internal components, allowing them to be swapped with minimal effort when necessary. The main body consists of lightweight materials including a plywood

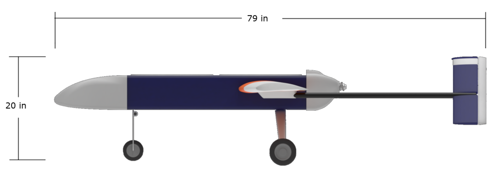


Figure 2: Left Side View of the Aircraft

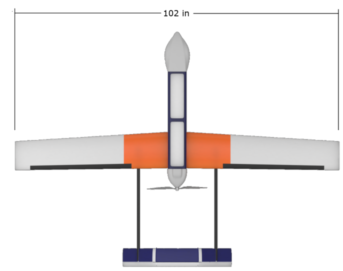


Figure 3: Top View of the Aircraft

fuselage frame with balsa paneling, carbon fiber beams supporting an inverted V-tail configuration, and a MonoKote skin to provide an aerodynamic outer coating. The inner structure employs a plywood truss design with specific attention towards bearing the lift of the wings (translated through a central aluminum beam and fiberglass supporting rod) and support points from the wheels, as well as the horizontal thrust and vibrations from the rear propeller. This truss layout translates applied forces evenly throughout the body to ensure stability during flight and flight maneuvers such as takeoff or landing. All internal components and their respective wires are secured to the rigid framework with adhesive material, screws, or Dual Lock in order to fully mitigate unstable movements during turbulence.

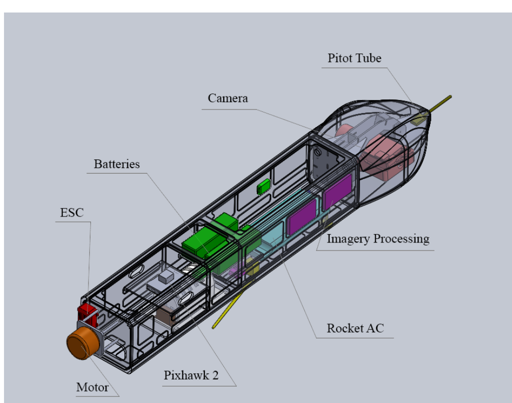


Figure 4: Internal Component Layout

2.1.2 Payload

The payload of the entire aircraft is divided into four compartments – nose cone, anterior fuselage, posterior fuselage, and motor housing. The nose cone compartment houses a digital airspeed sensor and the image recognition system hardware which includes the camera and gimbal setup. The nose cone was custom-made to be optimized around the payload within its compartment. The anterior fuselage compartment houses imagery processing system and the front landing gear system. The front landing gear is directly attached to the bulkhead. A servo is attached adjacent to the landing gear, providing operation of the front wheel. Rocket AC, Raspberry Pi, and the gimbal controller are all secured inside the anterior fuselage using Dual Lock. The posterior fuselage compartment holds the air delivery module, Pixhawk 2, and main and supplemental batteries. After multiple mass property analyses using Open VSP, the air delivery module was placed on the center of gravity (CG) of the plane to ensure that CG location is constant after the water bottle release. The motor housing compartment houses the motor for the rear propeller, securely held to the plywood firewall to prevent vibrations or precession during operation. An electronic speed control allow users or programs to modulate its effective thrust as needed and records voltage, current, and temperature.

2.1.3 Propulsion

The motor is a rear-running Scorpion SII 520KV with a 14x7 propeller. The propeller and motor combination were based on a data chart provided by the manufacturer listing the thrust efficiencies of various propellers given a 6s lithium polymer battery. The propeller is chosen with takeoff specifications in mind, in part due to the rear skids on the tail. Wing selection was based on projected weight and velocity; an estimate that stems from experience in previous years. A sample of the table we built is provided below in Table 2, which shows that given a $C_{L_{max}}$ of 1.2, stall speed of 12 m/s, and mass of 9 kg, the aircraft would need a planform area of about 0.8 square meters. However, once the aircraft was fully assembled, the mass fell to 6.8 kg, which led to a stall speed of 9 m/s, matching the stall speed recorded in testing of 8.5 m/s. The difference is probably due to underestimating the $C_{L_{max}}$ of the wings, since a Clark-Y airfoil has a $C_{L_{max}}$ of 1.3-1.4. [4]

2.1.4 Avionics

The avionics system on the UAS has been completely overhauled since the 2016-2017 season. Though the core design remains the same, all of the wires have been removed and redesigned with the whole system in mind to optimize wire lengths and minimize clutter. New connectors were also added for ease of assembly on the flight-line. LED switches as well as a new failsafe system featuring a deans plug make interacting with the plane and cycling power much more intuitive. The previously used Odroid-H.264 configuration has been replaced with a Raspberry Pi to control the ground communication system. The power system is triply redundant, allowing for up to two points of failure before power to the control surfaces are lost.

The batteries were chosen based on the needs of the system, projected flight time, and battery testing previously recorded by the Air Systems Lab at The University of Texas at Austin. Two 2200 mAh,

	Mass (Kg)	6	6.5	7	7.5	8	8.5	9.5
Wing	.825	9.46	9.84	10.21	10.57	10.92	11.25	11.90
Area	.85	9.32	9.70	10.06	10.42	10.76	11.09	11.72
(meters)	0.875	9.18	9.56	9.92	10.27	10.60	10.93	11.55

Table 2: Stall speed in m/s for a given mass and wing area

Mass	Fuselage Length	Fuselage Volume	Flight Time (Endurance)	Tail Boom Length	Climb Rate
6.8 kg	1.1303 m	.70 m^2	40 minutes	1.22 m	3.3 m/s

Table 3: Plane's overall flight metrics

3s ThunderPower (TP) Lipos were selected for the primary avionics and a 1350 mAh, 3s TP Lipo was chosen for the secondary avionics. The propulsion system runs on 2 5000 mAh, 6s TP Lipos, which is the limiting factor for endurance at 40 minutes. This was intentional, as the avionics should always outlive the propulsion system in case a gliding landing is required.

2.2 Autopilot

The UAV uses a Pixhawk 2 Flight Controller and runs PX4 autopilot. Its features consist of autonomous takeoff, flight, and landing. Moreover, it allows for automatic and manual tuning of the aircraft's PID values to improve its overall stability while flying autonomously. The Pixhawk 2 was suggested by the team safety pilot because of its robust structure and reliability. In order to communicate with the aircraft, the Pixhawk 2 allows the team to send and receive commands over multiple telemetry connections, e.g. both telemetry ports and the USB port, abilities not offered on the Pixhawk 1 and APM board. For GPS, the Here GPS and Magnetometer module were chosen, since both sensors benefit from being mounted on top of the aircraft. This was updated from the previous 3DR GPS and has the benefits of containing an arming switch and a redundant mounting plate. The I^2C pitot tube is mounted through the nose cone where it samples free stream uninterrupted airflow. The Pixhawk 2 requires two power sources: one for the servo rail and one for the board itself. Redundant power is used for each, with both the primary avionics and propulsion batteries providing power to the Pixhawk 2 and the primary and secondary avionics powering the servo rail. The secondary avionics battery is used exclusively for the servo rail and the RC receiver. The benefit of this is that if a short occurs elsewhere in the avionics circuit, the aircraft can still be landed by the safety pilot. Moreover, powering the Pixhawk 2 via primary avionics and propulsion batteries measures voltage and amperage use, which accurately assesses the remaining flight time, and determines aircraft landing.

2.2.1 Ground Station Software

The ground station has two windows. The first is Mission Planner, which is used for general flight purposes, such as flight planning, parameter tuning, and uploading new firmware. The second is called Flight View, which contains the information required for the judges, allowing the Mission Planner pilot to focus on the task at hand. The window displays waypoints, obstacles, no-fly zones, altitude, airspeed, map, and aircraft location. Flight View is served up from an in-house server and also has view for image overlay and flight history. Mission Planner allows for seamless configuration of tuning parameters and functions as a medium for issuing path commands to the aircraft. In addition, Mission Planner provides an intuitive GUI with crucial information such as altitude and battery percentage.

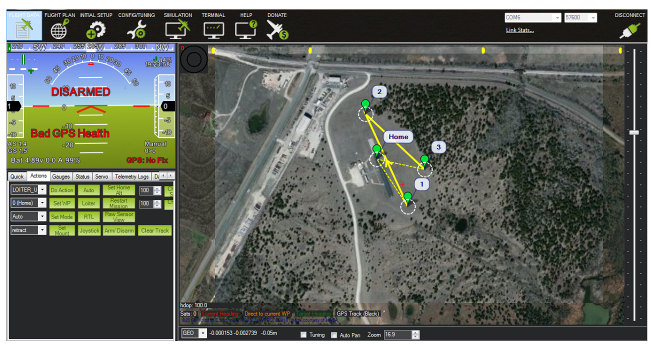


Figure 6: Mission Planner Home Screen

2.3 Software Systems

As opposed to building traditional monolithic programs, the UAV Austin software team built the software using a microservice architecture (MSA). With this pattern, the software is split into small individual services which communicate with one another with via HTTP servers and clients. Because of the additional overhead associated with creating microservices, developers are forced to take a more active approach at structuring their software.

Because HTTP requests are platform-agnostic, services can be written in different programming languages. This allows for a service to be written in a language that is most applicable to the task. Each service is containerized with Docker, which allows for a service to run on different machines with minimal configuration. This also makes the development environment similar to production. Both tests and missions are managed with Docker Compose.

2.3.1 Fault Tolerance

In theory, all errors can be handled with basic control flow logic. However, in practice, the sheer amount of edge cases that can occur make this task impossible. Designing software around the idea that it can fail, and that failures are expected, allows for the creation of

Wing Airfoil Type	Wing Planform Area	Wing Span	Taper Ratio	Leading Edge Sweep	Aspect ratio	Wing Loading
Clark-Y	.841 m^2	2.6 m	.6721	6.2 deg	8.0196	8.085 kg/m^2

Table 4: Wing Dimensions

more robust software. If a piece of software gets into an unexpected state, it should immediately exit. In production, all services are set in Docker Compose to restart unconditionally. If an error occurs, the services will restart and form a stable configuration again.

2.4 Obstacle Avoidance

The autopilot team decided on the A* search algorithm to accomplish obstacle avoidance due to its simplicity and speed. The field is first converted into a discrete map of nodes to perform A* on. The Jump Point Search (JPS) algorithm was initially considered due to its superior speed to A* in the case of homogeneous grids, but it was ultimately decided against due to its additional complexity.

2.4.1 A* algorithm

The two common path-finding algorithms are breadth first search and greedy first search, though each have their downfalls. Breadth first search is guaranteed to find an optimal path but exploring all neighboring nodes from source until a destination is reached. However, this requires unnecessary exploration of extra nodes and reduces performance. Dijkstra's algorithm improves on this to allow edges of differing weights, but the same problem of checking poor paths persists, and proper weighting is difficult. On the other hand, greedy first search attempts to reach the destination as soon as possible by using a heuristic value to estimate distance to the target. While greedy first search is typically faster, it is not guaranteed to give the optimal path.

A* combines the breadth first search and greedy first search to achieve the accuracy of the former and speed of the latter. Such is accomplished by representing the cost as a sum of the traversal cost and the estimated distance heuristic. As such, the nodes along a straight line path assuming there aren't obstacles will always be those with the lowest cost. At each iteration, the unevaluated node with the lowest cost is processed first, which allows the algorithm to advance toward the end, similar to the greedy search algorithm.

2.4.2 Algorithm Implementation

The autopilot team settled on using the Rust programming language for its safety and speed. The A* algorithm first generates a node map using the fly zone and obstacle data and stores it in a PathFinder object. Then, the PathFinder object calculates the optimal path using A* and returns the result as a list of waypoints. Similar to the rest of the software team, the path finding library is accessed through a microservice, which converts the data to the appropriate object.

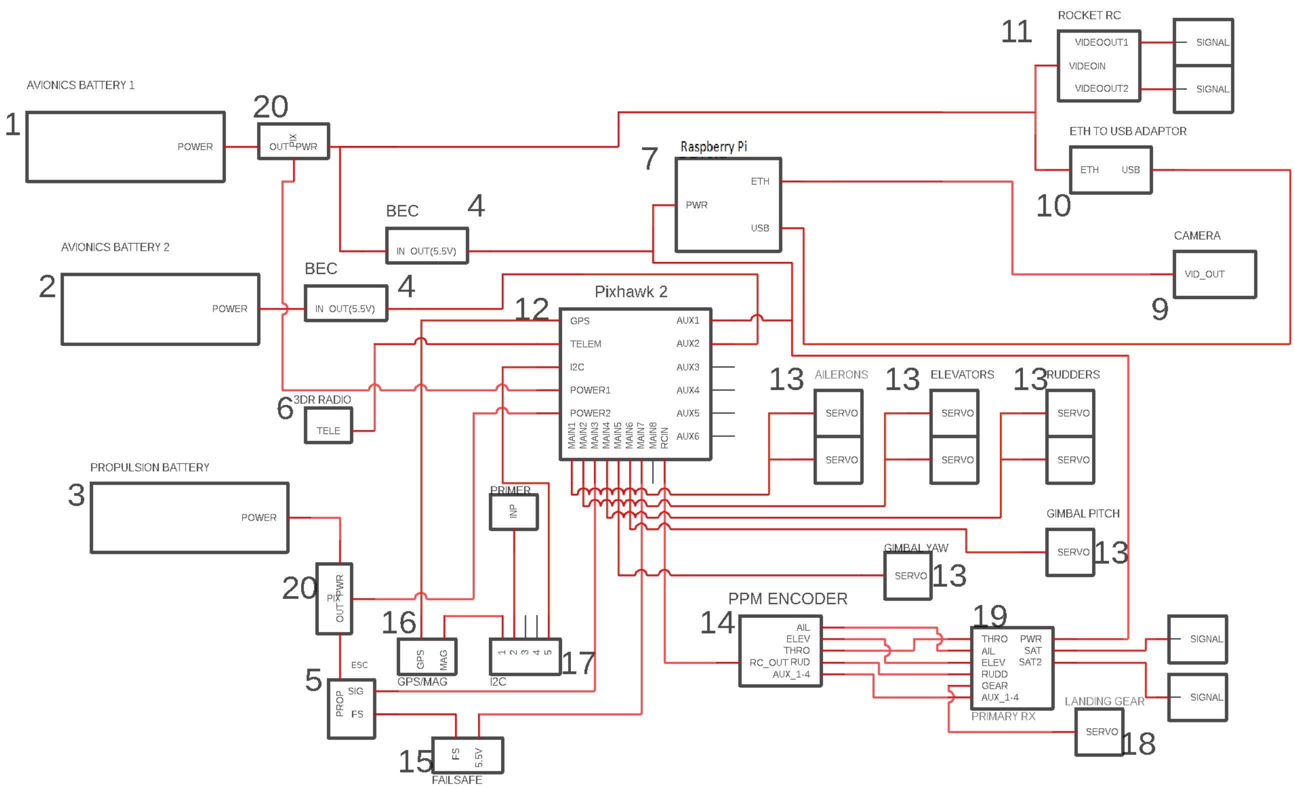
The node map is generated primarily through the haversine formula, which determines the distance between two points on a sphere based on their latitude and longitude. When a PathFinder is initialized, a fly zone of at least three points is required. The lowest latitude and longitude of the fly zone is subsequently used as the origin. Using the origin, any coordinate pair within the fly zone can be converted into a node with x-y coordinates using their haversine distance to the origin and a preset value for the size of a node, which is default to a meter long square. Furthermore, the program generates the outer bounds of the fly zone by drawing a line between the points of the fly zone and adding nodes along the line to a hashmap representing obstacles. Stationary obstacles are added by finding all nodes along the edge and pushing them to the obstacle list.

2.4.3 Moving Obstacles

Moving obstacles were not integrated into the A* implementation due to the complexity of passing this objective. Instead, the Mission Planner operator may manually move waypoints in order to avoid these obstacles if necessary.

2.5 Imaging System

The Sony Alpha a6000 with the E PZ 16-50 F3.5-5.6 OSS lens was chosen as the camera for the imaging system. This mirrorless, digital camera provides high-resolution imaging capability of 24.3 MP and 3.1x zoom. The a6000 utilizes an APS-C sensor, which has an area 1.6 times larger than that of a 4/3 sensors, resulting in detailed production of images. Additionally, the BIONZ X processor on the camera allows the system to accurately capture texture and reduce blurry details. Moreover, the camera features Fast Hybrid Autofocus with reduced reliance on contrast detection and the ISO range of 25600. Ultimately, Sony Alpha a6000 has the ability to produce high-quality images independent from altitude and lighting conditions. In comparison with last year's camera (Sony FCB-EV7100), the Sony Alpha a6000 has a significantly higher image quality (2.4 MP with 10x zoom vs 24 MP with 3.1x zoom) and comparable shutter speed. This is significant as the limiting factor for UAV Austin's image recognition system in the previous year was the resolution of the images provided by the team's camera. The size of a6000 is larger in size than FCB-EV7100 with its physical dimensions of 4.72 in $\times$ 2.63in $\times$ 1.78 in. However, this fact was outweighed considering that the camera's imaging abilities are comparable to a conventional DSLR while proving to be more compact than them. The mission required at least 40 pixels of width and 40 pixels of height on the target, which is two feet in length. with 313.775 ft as the maximum flight altitude and 11 m/s as the flight speed, the pixel dimensions of the image captured by the camera met the requirement. Also, the said flight speed and maximum flight altitude, the swath width of the camera is 142.945 ft, proving the a6000's effectiveness regarding the missions of this year's AUVSI SUAS competition.



Legend

1. Primary Avionics Battery	2. Secondary Avionics Battery	3. Propulsion Battery	4. Avionics BEC
5. Electronic Speed Control (ESC)	6. 3DR Radio	7. Raspberry Pi	8. N/A
9. Camera	10. Ethernet to USB Adapter	11. Rocket AC	12. Pixhawk 2
13. Servos	14. PPM Encoder	15. Throttle Failsafe	16. GPS and Magnetometer
17. I2C Splitter	18. Landing Gear	19. Primary Rx	20. Power Brick

Figure 5: Wiring Diagram

Figure 7: Sony Alpha a6000, Onboard Camera

2.5.1 Imaging System Hardware

For the 2018 competition UAS, a complete redesign was done on the imaging system hardware due to the change in physical constraints that came with the newly selected camera. The dimensions of Sony a6000 are more restrictive compared to the previous year's camera. The width of the camera proved to the most constraining dimension. In previous years, the imaging system hardware was housed within the fuselage of the UAS, for it was the most convenient and conventional to do so. However, the team analyzed that housing the imaging system hardware within the fuselage would restrict the viewing angle of the camera drastically unless significant drag-inducing modifications were to be made to the fuselage. Therefore, the team came to the decision to locate the payload within the nose cone of the UAS. Imaging system hardware for this year's competition UAS consist of two main parts – the camera and the gimbal. The prime requirement for the gimbal set by the team was for it to provide the necessary viewing angles for the camera. The gimbal comprises of two sets of brackets and brushless gimbal motors, and an arm structure that secures the gimbal to the bulkhead of the nose cone. Also, bearings were incorporated into the design to guarantee smooth angular movements. In order for the imaging system hardware to be housed within the nose cone, the nose cone itself was required to be redesigned as well. The two requirements for the nose cone was to adequately house the gimbal and the camera without restricting their viewing angles while minimizing drag. Multiple iterations were designed for the team to have the optimized imaging system hardware. The first iteration was manufactured using the method of rapid prototyping. All parts were printed using ABS to ensure structural integrity. The first iteration provided the necessary viewing angles; however, it was not optimized spatially. Also, the use of ABS as the primary material resulted in a relatively heavy payload. On the processing side, the team decided to use specialized hardware (a Jetson TX2) to accelerate the processing of potential target images. This allows a greater degree of autonomy for the image recognition process (as this provides the throughput needed to process all images received - no manual pre-filtering required) and simplifies the overall image processing setup.zz

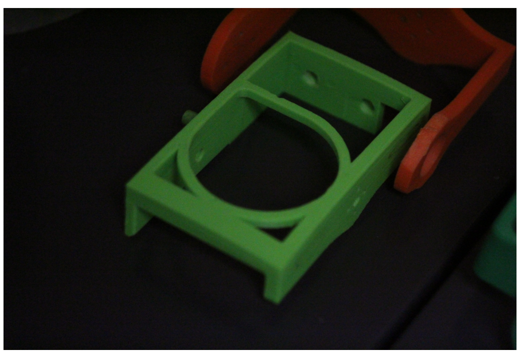


Figure 8: 3D Printed ABS Bracket, Iteration 1

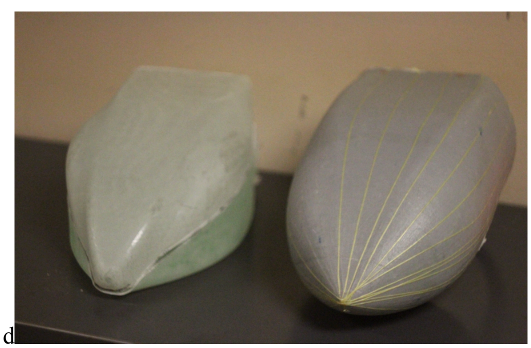
d


Figure 9: Iteration 2 (left), Nosecone Iteration 1 (right)

The final iteration of the imaging system hardware keeps the mechanism of the previous iteration while prioritizing spatial and mass optimization. The gimbal brackets are printed in carbon fiber-reinforced high strength nylon. Therefore, the overall thickness of the brackets were reduced. Also, the arm structure is composed of laser-cut plywood parts, which decreased the overall mass of the system while keeping the structural integrity. Lastly, the nose cone was redesigned to reduce unnecessary space inside the structure. Computer simulations were also done to ensure minimization of drag caused by the nose cone. The nose cone is composed of fiberglass to decrease the mass and increase the structural integrity.

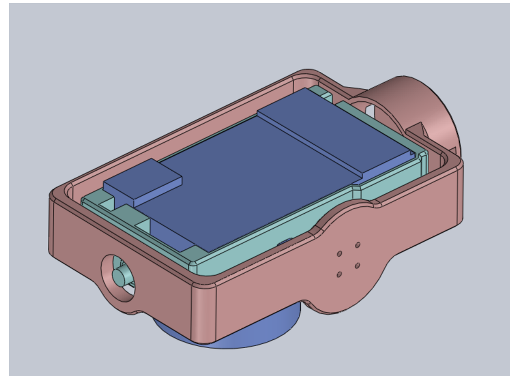


Figure 10: Final Iteration of Gimbal System

2.6 Object Detection, Classification, Localization

Before images are processed they are warped to true shape if possible using the telemetry at the time the image was taken. During normal operation, the camera points downwards in flight. However, due to variations in the orientation it becomes helpful to warp them to make target identification and location easier. A bicubic interpolation was used in OpenCV on board the plane to accomplish this with transparency in the corners after. North in every warped picture also points straight up in the new image. Images will not be warped if the horizon if visible. In most cases this should not be a concern, unless a heavy angle is needed for the off-axis target, then the original image will simply be used.

2.6.1 Preprocessing

Given the unwarped aerial image, there are several image transformations that are performed sequentially to optimize potential flaws in image quality introduced by the camera. The primary goal of the preprocessing phase is to ensure that all actual shapes are properly detected as blobs (portions of the aerial image that are potential shapes, but have yet to be classified), and that blobs are indeed good representations of potential shapes and not debris, patches of grass, or other noise. Ideally, preprocessing will result with blobs that are only shapes, and their characteristics are yet to be determined. However, the classification phase is robust in determining blobs that are not actual shapes. Hence, preprocessing attempts to minimize the number of blobs detected that are not actual shapes, so as to reduce the number of classifications that need to be performed, thereby reducing the computational time necessary.

The sequence of image transformations begins with performing a canny edge detection with variable thresholds to account for potentially blurry images. If, for a given blob, the resulting number of connected neighboring pixel sets do not exceed a preset upper bound, the edges are further refined using the varied threshold to remove excessive noise. The resulting edges are tightly bounded and are each standardized through a combination of region filling, dilation, and further noise filtering. These transformations can mitigate certain distortions such as discontinuities that are common for small targets. Finally, the preprocessing is complete when each blob is cropped out of the original aerial image, thus preserving color details that are important for classification.

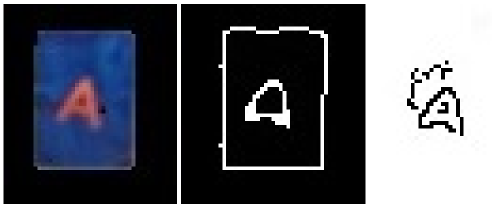


Figure 11: Masked Image, Shape Edge Detection, and Alphanumeric Edge Detection

2.6.2 Classification

The classification phase involves determining all of the characteristics of a target. Shape detection is performed with a retrained version of Google's deep neural network model Inception-v3 using TensorFlow. The model consists of several computational layers, such as convolution, max pooling, and softmax. The original model's final classification layer is retrained with a dataset of training images, consisting of at least 10000 synthetic images for each of the thirteen shapes: circles, crosses, heptagons, hexagons, octagon, pentagon, quarter circles, rectangles, semi circles, squares, stars, trapezoids, and triangles. After cross-validation, the resulting training accuracy is around 93%. [5]

Each blob, defined by its corresponding bounding box, is classified with this pre-trained model, which outputs a shape classification and confidence score. If the confidence exceeds a given threshold, the shape's pertinent information is appropriately saved and the blob proceeds to alphanumeric detection. Alphanumeric detection (OCR) is performed on the edges found within a confirmed shape using `pytesseract`. The alphanumeric detection attempts to rotate the blob with fixed increments, and determines the highest confidence among all rotations, thereby assigning the alphanumeric value and the corresponding rotation of the shape. Lastly, a black mask is placed around the target to eliminate any background colors. The colors of the shape and the alphanumeric are determined using k-means clustering to identify the two main RGB values in the target. These values are then passed through an algorithm to find the colors with the closest predefined RGB values. The more dominant of the two will be the shape color, while the other will be the alphanumeric color.

2.7 Communications

The aircraft must be able to transmit and receive on both the 900 MHz, for Pixhawk 2 telemetry, and 5.8 GHz band, for Imagery related telemetry, throughout the whole mission, plus receive only on the 2.4 GHz band for manual control. To test this, two senior design teams powered on all of their systems right next to the ground station with the aircraft out on the runway. While there was no drop in the lower two bands, the 5.8 GHz signal dropped completely. To remedy this, the signal bandwidth was reduced from 80 MHz to 20 MHz, meaning that the noise generated by the analog transmitters could be avoided more easily by switching to another frequency in the 5.8 GHz band. A direct result of this is a loss in data rate, but there was no visible drop in video quality. As a result, the UAS is prepared for the noisy environment out at Webster Field. As a side note, if everyone is on WiFi then there should not be a problem at 80 MHz, since WiFi is regulated to avoid jamming itself whereas analog video antennae tend to bleed over all the channels. Conversely, analog video systems will likely have many problems out at the field.

In the previous year's competition, the communications system was not able to communicate successfully with the interoperability server. Some packets were being delivered correctly, however, not enough were being sent to allow for ODLC submission. Because of the nature of networking problems, it was not possible to reproduce the exact scenario. However, as a preventative measure, the router for UAS was replaced with the ASUS RT-N66U. The ASUS RT-N66U is a gigabit router which allows for the installation of custom firmware.

2.7.1 Communications Monitoring

To prevent a communications dropout for this competition year, it is required to have runtime analytics to know if an issue is arising. For this purpose, a communications dashboard was built to display the current ping of different runtime services and the interoperability server, and the rate at which telemetry is being uploaded to the interoperability server. The telemetry upload rate is divided into two types: the raw number of telemetry being sent, and the amount of unique telemetry being sent. Though uploading duplicate telemetry does not count towards the telemetry rate, it is useful to ensure that the communication link with the interoperability service is stable. Figure 12 shows the terminal dashboard.

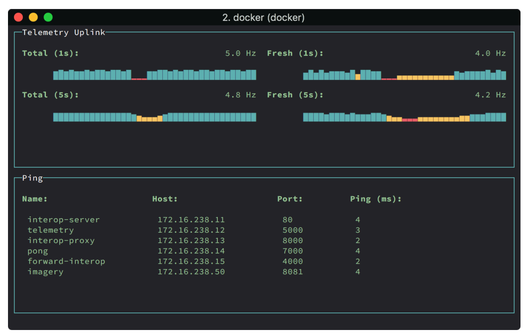


Figure 12: Communications Dashboard

2.7.2 Protocol Buffers

In the previous competition year, one of the things the UAV Austin software team struggled with was effectively communicating between different pieces of software in the software stack. Though there were somewhat defined schemas for the datatypes and endpoints, much time and effort was spent navigating the networking libraries, serialization/deserialization routines, and parsing tools of the languages used in the stack. This year, especially with the move to a proper microservice architecture, it was deemed prudent to use a standardized, self-documenting, platform independent data serialization method: namely, Google Protocol Buffers. These have enabled the software team to seamlessly communicate between the various services in the stack even though they are written in different languages. Additionally, datatypes can be added and modified without sacrificing backwards compatibility and datatypes are cleanly separated from application logic.

2.7.3 Interoperability Server Communications

Because all the services communicate over Protocol Buffers, a service was created to wrap the interoperability server's endpoint with Protocol Buffers. This service, *interop-proxy*, acts as the front lines for the interoperability service communications. Because this service must remain reliable, *interop-proxy* was written using the Elixir programming language. Elixir is a language built on top of the Erlang VM. Erlang has its roots in telecoms, and is used by companies such as WhatsApp to make very resilient communications systems. [1]

2.8 Air Delivery

The air delivery module has gone through multiple iterations from last year's unit. The air delivery system from last year proved to be effective at securing and releasing the payload in both automatic and manual modes. However, the team found two main areas for improvement – overall mass and delivery mechanism. Last year's delivery system utilized aluminum as its main material for its overall structural integrity. This resulted in a relatively high-mass unit, increasing the overall weight of the aircraft. Secondly, the delivery mechanism was driven by a servo unit and multiple connection pieces. This increased the number of possible failure points. Also, the delivery system required the servo to complete 90 degrees of rotation. This restricted the system from making an instant drop.

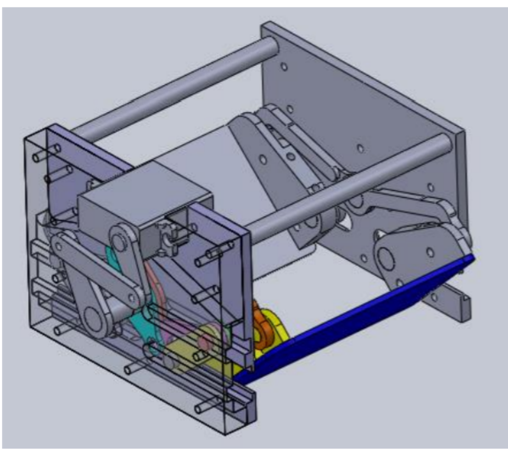


Figure 13: Air Delivery Module of 2017 System

The first iteration of the new delivery unit featured an external payload design. The capsule which enclosed the water bottle were placed on the external surface of the aircraft. The capsule and its payload were secured in place by the actuated servo horn. The payload was released as the servo executed 15 degrees rotation, inducing a near-instant drop. This system proved to be lightweight as it consisted of laser-cut plywood and plastic-coated industrial paper. This iteration was done with the sole purpose of mechanism testing as the team sought viability in a simpler delivery mechanism. Therefore, the unit fell short in its implementation viability as it would have increased the overall drag of the aircraft due to its semi-submersible design.

The final iteration is a revised version of the external delivery unit from the previous iteration. It was modified to feature an internal payload. The structure was built using 3D printing technology. With the exceptions of few special components such as the servo and the hinge, the system is composed of PLA, allowing the unit to be lightweight. The system includes a simple mechanism which holds and releases the payload secured inside the enclosed structure. The mechanism is minimalistic, drastically decreasing the possibility for system failure. The entire unit is placed inside the fuselage of the aircraft, which eliminates the previous problem regarding increased drag.

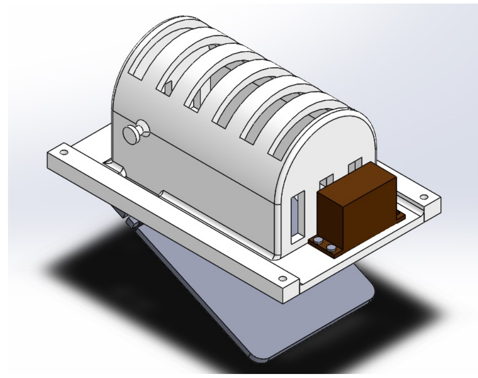


Figure 14: Final Iteration

2.9 Cybersecurity

With the increasing popularity of unmanned aircraft systems in both civilian and military applications, they have become a lucrative target for malicious agents due to the immaturity of the technology.

To understand the security risks of operating the entire UAS, both physical and remote attack surfaces must be evaluated. For the purposes of the competition, the likelihood of an attack is extremely low, but the overall impact of a potential attack is high.

2.9.1 Physical

Physical attack surfaces include all physically accessible equipment that interoperates with the UAS for direct flight control. In accordance with conventional server and network security, it is assumed that if physical access to equipment is gained, the equipment in question is assumed to be entirely compromised. Vulnerable physical equipment is listed in Table 5.

Component	Attack	Mitigation	Fallback
Ground control laptop	Physical takeover	The laptop is physically protected by multiple personnel.	Perform a manual landing.
Imagine processing worker laptops	Takeover (physical or remote)	Run a minimal Linux distribution solely for image processing.	Designate another machine for image processing.
Fuselage and onboard computer	Sabotage	Supervise the aircraft. In the future, use Torx security screws to secure the fuselage.	None possible

Table 5: Physical Attack Surfaces

2.9.2 Remote

Remote attack surfaces include all equipment that can communicate over a radiofrequency.

As autopilot communications may operate over either Wi-Fi or 3DR, successfully compromising the aircraft requires both links to be hijacked; otherwise, the untrusted link can be cut to restore legitimate control of the aircraft.

Additionally, gaining access to the network, which would require getting past WPA2 or gaining physical access to the router alone, does not compromise the aircraft as the only mission critical component that can be accessed over the network is the Pixhawk

Component	Attack	Mitigation	Fallback
Ground control laptop	Remotely (over network)	Harden the OS, remove unnecessary software, and disconnect the laptop from the Internet.	Perform a manual landing.
GPS Receiver	Spoofing / Jamming	u-blox receivers feature built-in spoofing detection to automatically trigger failsafe mode in such an event. Receiver is oriented upward and sits on a metal plate to minimize interference.	Switch to return-to-land or loiter mode.
3DR Link	Sniffing	Link is secured using AES and frequency-hopping spread spectrum (FHSS).	None possible.
3DR Link	Sniffing	Link is secured using AES and frequency-hopping spread spectrum (FHSS).	None possible.
3DR Link	Interception	Link is secured using AES and frequency-hopping spread spectrum (FHSS).	Attempt manual landing, or disconnect link and force RTL.
3DR Link	Jamming	Link operates on an FCC-regulated frequency (900 MHz).	Use Wi-Fi until jamming abates.
RC receiver	Interception	Receiver and transmitters are paired before the competition; FHSS prevents hijacking without knowledge of initial seed.	Switch fight control mode to telemetry link only.
RC receiver	Jamming	Receivers are placed redundantly on both wings to minimize interference.	aircraft returns to land after 30 seconds.
Wi-Fi Link	Sniffing	Access point is secured using conventional WPA2 encryption and is patched against KRACK. All network endpoints are password-protected.	None possible.
Wi-Fi Link	Interception	Access point is secured using conventional WPA2 encryption and is patched against KRACK. All network endpoints are password-protected.	Perform a manual landing.
Wi-Fi Link	Jamming	Transceiver automatically switches to lowest-noise frequency in the 5.8 GHz band, and it uses the least amount of bandwidth (20 MHz).	Perform a manual landing.

Table 6: Remote Attack Surfaces

connected over serial to the onboard computer and access to it requires gaining access to the onboard computer which is secured by an RSA key and exposes only the necessary TCP/IP ports.

Furthermore, the software stack utilizes container based microservices which run Alpine Linux, a minimal base image with reduced attack surfaces, have only the necessary ports publicly accessible, and are isolated from the rest of the system. While some of these services do expose public endpoints that would be accessible if an attacker were to gain access to the network, none are mission critical. Thanks to the fault tolerant nature of microservices any service that crashes will restart harmlessly.

2.10 Testing

Missions and systems testing were planned and executed on a rolling schedule. Each test outlined the expected tasks and gradually accomplished the steps of the mission, from initial preparation to final mission and packing. The team performed comprehensive tests to simulate and evaluate autonomous flight, waypoint navigation, air delivery, imaging systems and analysis, server interaction, obstacle avoidance, and water dispersion upon impact. During full mission tests, designated observers from the team documented errors or delays within the aircraft, mission, and team efficiency. Once this was found, the hardware team worked to improve the unsatisfactory systems. Furthermore, should a system under perform, it was deemed unreliable and underwent various testing analysis. A system is not competition ready without constant and thorough testing iterations until the UAS reaches ideal conditions.

Testing was performed successfully, however, the schedule was delayed by the enhancement of the team's 2017 AUVSI SUAS competition aircraft, and design project to develop a fiberglass aircraft, all of which reinforced the team's hardware skills and design expertise. Furthermore, the organization created a research and development team that worked closely with University of Texas researcher and systems engineer, Dr. Armand Chaput to design the new aircraft. The chart below describes some of the design improvements the team was able to make through the systems and flight test processes.

For software testing, a test suite was created with Docker Compose which ran a simulated plate so that the entirety of the software microservices could run without depending on the Pixhawk directly.

3 Safety, Risks, and Mitigation

The primary task of UAV Austin is to operate safely and mitigate all possible risks, the solution to this is to develop a plan for all possible anomalies and create procedures/checklists to ensure success. For UAS design and development, a Gantt chart must be used to make sure that critical deadlines are not missed. For flight tests, checklists and procedures are maintained and followed for flight preparation, packing, transportation, setup, and takedown. A flight plan is created for each flight for both pilot, and ground crew to ensure all objectives are met for each flight. If objectives are not met, the anomalies must be documented so that they can be mitigated before the next flight.

3.1 Development Risks and Mitigation

Table 8 defines the consequences of the potential risks listed in Table 9.

Feature to Improve	System Issue	Method of Improvement
Weight Distribution	Center of gravity located too far in the rear of aircraft	Reworking of electronics and nose cone in order to balance center of gravity
Gimbal	Too bulky, needed a larger range of motion	Redesign with attention to size and range of motion
Nose Cone	Not accommodating of camera, too heavy, could be more aerodynamic	Complete redesign with aerodynamic characteristics in mind, component constructed with lighter materials and reinforced with Kevlar to increase strength without compromising weight
Camera	Previous camera underperformed in terms of resolution and shutter speed	Research and selection of new camera In order to improve overall range and resolution of operation
Landing Gear	Lack of range of motion and stability	Relocation of landing gear-mounted directly to front bulkhead to ensure durability,sandwiched the front landing gear with washers so that the inner diameter of the washers is closer to rod diameter

Table 7: Refinements Made via Testing Process

Consequence	Detail
Catastrophic	Extremely long schedule setback (2+ weeks); Permanent damage to equipment; Personnel or third-person injury
Critical	Long Schedule setback (1-2 weeks) ; Structural damage that does not allow for immediate flight; low-medium degree of personnel injury
Significant	Schedule setback (3-6 days); small equipment damage (still able to fly); low degree of personnel injury
Marginal	Relatively short schedule setback (0-2 days); very low-degree equipment damage (can be fixed immediately)

Table 8: Definition of Consequences

3.2 Mission Risks and Mitigations

Table 10 contains a list of mission critical risks and how the team intends to mitigate them. Although no plan gets executed perfectly, UAV Austin is confident that it can contain pressing risks respond quickly to any issues that arise.

4 Conclusion

During the past year, UAV Austin has dedicated its efforts to designing, optimizing, and testing the UAS in preparation for the AUVSI SUAS 2018 competition. The year involved a complete redesign of internal structures, image recognition, system hardware, nose cone, air delivery module, and significant improvements on the landing gears. The redesign and modification was followed by extensive full mission testing to ensure reliability and task completion. UAV Austin has gained a comprehensive understanding of systems engineering to implement new changes required for the competition. UAV Austin has also redefined its team philosophy. As a team, UAV Austin focuses on the technical product as well as the education and growth of its members in order to ensure the longevity and sustainability of the group. The philosophical shift was prompted by the understanding of the impact that young members have on the team. Their commitment and passion for learning has deepened the team's purpose to include detailed design of a long-term plan for UAV Austin. UAV Austin is not only optimistic about this year's AUVSI SUAS competition with Sir-Vay-Lance, but also looking forward to the continued development and improvement of long-term technical plans.

The University of Texas at Austin Unmanned Aerial Vehicle team enthusiastically awaits the opportunity to participate in the 2018 AUVSI SUAS competition in order to gain experience and to display the vision of UAV Austin: to turn ideas into innovations.

References

[1] Katelyn Bogucki. Erlang: From WhatsApp To Outerspace". `https://www.huffingtonpost.com/2014/03/11/erlang-from-whatsapp-to-o_n_4944667.html`, March 2014. [Online; accessed 4-May-2018].

[2] AUVSI Seafarer Chapter. Competition Rules | SUAS 2018. `http://www.auvsi-suas.org/static/competitions/2018/auvsi_suas-2018-rules.pdf`, October 2017. [Online; accessed 4-May-2018].

[3] INCOSE. What is systems engineering? `https://www.incose.org/AboutSE/WhatIsSE`, 2018. [Online; accessed 4-May-2018].

[4] Daniel P. Raymer. *Aircraft Design: A Conceptual Approach, Fifth Edition*. 2012.

[5] TensorFlow. Image recognition. `https://www.tensorflow.org/tutorials/image_recognition`, 2018. [Online; accessed 04-May-2018].

Risk	Description	Consequence	Mitigation	Fall Back Plan
Loss of Vehicle	Complete loss of the UAS system due to unforseen mechanical or communication anomalies	Catastrophic	Ensure redundancy in the communication system, and have a QA check the mechanical attachment points and linkages in the vehicle before flight	Re-outfit an old senior design aircraft with the internals and re-preform integration testing including HITL testing for software
Not enough funds to finance the UAS	Cannot buy critical components later in year	Catastrophic	Track expenses in an Excel document	Fundraise for more money
Design/Manufacturing delays	Critical aircraft hardware/software is not finished on time due to unforseen circumstances	Critical	Set preemptive deadlines and perform regular checkups, especially on crucial projects.	Find alternative solution. Focus on functionality over complexity or aesthetic.
Aircraft is overweight	The aircraft's weight exceeds estimations, which negatively effects its flight performance	Critical	A mass properties Excel sheet tracked the mass and location of all components and their wiring.	Optimize the motor and propeller size to fly at a higher speed.
Electrical Mal-function/Damage	Exposing electronics to excessive voltage or current can damage them.	Critical	Read each components' documentation to ensure the correct voltage is supplied. Follow a circuit diagram for installation. Ensure all wires are properly insulated, and all connections are properly crimped.	Have backups of flight critical electronics such that flight tests may continue if a failure is to occur.
Inclement Weather	If there is inclement weather at the airfield we cannot proceed on schedule with the flight tests	Significant	Ensure that there is a margin on flight testing dates, such that if there is inclement weather the flight test can be moved and not cause significant delays.	Have a student pilot trained and ready for flight on a training vehicle such that regardless of scheduling a test flight may occur.
Center of mass (COM) is too far away from the quarter-chord of the wing	The aircraft's center of mass (CM) must be within a margin of error to the quarter-chord of the wing to ensure stable yet controllable flight	Marginal	A VSP model with all the internal component masses is kept up to date and ensures proper placement of the CM	Rearrange the internal components to balance the aircraft. Add ballast as a last resort

Table 9: Developmental Risks and Mitigations

Risk	Description	Mitigation	Fall Back Plan
Loss of Manual Control	The ground crew may lose control of the UAS by exceeding its max range or depleting the batteries. This may result in the aircraft failure	Range tests before flight. All battery voltages will be tested before flight. Ground crew will plan to land with a 20% battery reserve.	Safety Pilot will take control of plane. If safety pilot is unable to connect, plane will return back to home. If no connection is made within 30 seconds, the plane will enter fail safe mode.
Pitot Tube Failure	Pitot tube does not return accurate value for airspeed.	Pitot tube will be calibrated and tested before flight.	Change setting in the flight control software to rely on GPS for groundspeed.
Loss of WiFi Link	WiFi link is for telemetry and imagery	Use antenna tracker and ensure clear line of sight	Use Mavproxy to send telemetry data to Flight View. Download imagery from the UAS after flight for processing
Loss of 3DR Link	The primary link between Mission Planner and the autopilot	Maintain clear lines of sight with the UAS and use USB current booster	Use Flight View to tell UAS to return to home and wait until the primary link is regained. Try to reroute Mavlink to Mission Planner through the 5.8 GHz link with Mavproxy
Failing to meet the mission time limits	Failing to meet the 20 minute setup or 45 minute mission time limit.	Mission will be repeatedly rehearsed before the competition. Ground station will record the amount of time left for the setup and mission.	If the 45 minute mission time limit is about to be reached, cease all mission objectives land the plane immediately.
Wind Interference	High winds will alter the UAS's flight path and may cause it to miss waypoints or violate the no fly zone boundary.	The ground station flight planner will monitor wind direction and speed. They will adjust the flight plan to ensure the plane does not violate the no fly zone	If the no fly zone boundary is violated, then return inside the boundary as soon as possible. If a required waypoint is missed, adjust the flight plan to hit the waypoint, correcting for the high wind.
Air Delivery Mechanism (ADM) Malfunction	The air delivery mechanism (ADM) may malfunction, wherein the servo burns out or water bottle gets caught.	The ADM will be tested before flight to ensure that the paylaod does not get caught in the mechanism.	Perform the rest of the mission as usual.

Table 10: Mission Risks and Mitigations