2015 SUAS Competition Journal Paper

Abstract: The Mizzou Unmanned Systems Team, or MUST for short, is participating in the annual SUAS competition this year for the first time. We are a collection of students from a multitude of academic backgrounds ranging from Engineering to Arts & Sciences. We begin the paper with a description and justification of the chosen systems onboard the aircraft. We then "zoom out" to see the bigger picture and discuss the overall design of the UAS. We continue by presenting our testing methods, results, and conclusions. We expand on this to include our safety-oriented design choices. We conclude the paper with our affirmative opinion that MUST will meet or exceed our expectations for this competition year.

Section 1: Systems Engineering

Mission Requirements Analysis

Our primary mission objectives include autonomous flight and search area target recognition. Our secondary mission objectives include ADLC, actionable intelligence, off-axis target, emergent target, SRIC, air-drop, and interoperability.

Design Rationale

For the autonomous flight task, we knew that we would need an autopilot module (flight controller) capable of achieving programmable and autonomous waypoint navigation as well as the capability to support telemetry information to be transmitted down to the ground station. While it was not required, we also determined that adding a GPS compass would greatly increase our chances of successful waypoint navigation within the "objective" margin of error listed in the rules. Our team purchased a high-powered laptop to run the ground control station (GCS) software that would communicate with the flight controller via the telemetry link. We were each trained on using the GCS software by one of our senior team members, Will Bezold, who has had nearly five years of experience with the software and three years' experience participating in the SUAS competition. We felt this was necessary so that all team members were qualified to man the GCS software should the situation merit it.

For the search area task, we knew that the best weapon in image recognition is a clear, crisp image. Our team purchased a Canon EOS 6D and fabricated a custom bottom-facing mount for it to be placed in our aircraft. With a larger, cleaner amount of data to work with, a more confident output can be generated provided that there is enough CPU and GPU power to provide to our target recognition software. To that end, our team purchased a second high-powered laptop that is entirely dedicated to the image processing software. Our testing of the software has shown that all targets within an image are easily detected, and target classifications are accurately identified more often than not; our software seems to struggle most in identifying the shape of a target, but we are confident that we will have these issues ironed out before the competition. The QRC target is trivially found by our software based on the stark contrast between the QRC code and its surroundings, and is decoded in a matter of moments by the same software employing an open-source QRC decoding method. After the plane has landed and all images have been processed, the software takes the characters detected in legitimate targets and feeds them to an open-source "Scrabble" word generator that has been incorporated into the software. As our camera is downward-facing and not mounted on a gimbal, our flight controller will be programmed to fly the plane parallel to the ground as often as possible by making steep aileron/rudder turns for changes of course.

The above methodologies also pertain to the ADLC task. We note that, at the time of writing, our FAR is higher than 50%. We are working hard to isolate the issues and remedy them, but we are confident that we will be able to process the actual targets in time regardless.

The methodologies above also pertain to the actionable intelligence task, but with the additional element of the air-to-ground wireless network. Our telemetry link does not have the bandwidth required to reliably transmit our images at the same time, so we have implemented hardware to facilitate a wireless network on the 5.8GHz frequency spectrum. This hardware consists of a router coupled with a powerful directional beam-width antenna on the ground and an access point with two circular polarized whip antennas mounted to the belly of the aircraft. We power the access point in the air with a 12V battery pack and a DC-to-DC converter. The camera connects to an on-board computer whose sole task is to queue images for transmission down to the ground station. We use the 5.8GHz spectrum in order to keep our other frequency spectrums clear; we perform radio control over 2.4GHz, live video transmission over 1.2GHz, and telemetry over 900MHz.

For the off-axis target task, we plan on capturing images of the target while banking alongside the no-fly zone boundary. We are confident that the superior dimensions with which our camera captures photos will allow us to accurately determine the characteristics of the target without crossing over into the no-fly zone.

For the emergent target task, all members of the team have been trained on how to add a waypoint to a mission mid-flight. This will ensure that anybody will be able to add the waypoint location of the emergent target regardless of who may be in the best position to re-program the flight path.

For the SRIC task, we have created a process to run in the background on our aircraft's onboard computer. This process will lay dormant until it detects a device within range matching the IP address of the SRIC. The process will then drop the connection to the 5.8GHz image transmission network to connect to the SRIC. At the same time, the plane will travel in slow, spinning circles to ensure that the process has sufficient time to execute both the download and upload. Once the download and upload is complete, or after the connection to the SRIC is lost, the process will restore the connection to the 5.8GHz image transmission network. If the SRIC task was completed successfully, the process will stop. If the SRIC task was interrupted, the process will continue to lay dormant until it detects the SRIC again, at which point it will try to complete the task again.

For the air-drop task, we have mounted a high-powered servo to the bottom of the plane that is attached to a drop mechanism. This mechanism uses a spring-loaded retaining pin to hold the payload in place. When the air-drop is triggered, the servo will pull on a string which will retract the spring-loaded retaining pin from the drop mechanism, releasing the payload from the belly of the aircraft. One of our auxiliary radio control channels is dedicated to allowing and disallowing the air-drop servo to be moved; if our safety pilot has the "air-drop switch" held, the flight controller will be able to send a signal to the high-powered servo causing the egg to drop. If the safety pilot is not holding the air-drop switch, the signal that the FC sends to the servo will not produce sufficient servo travel to fully retract the retaining pin and drop the egg.

Finally, for the interoperability task, we have developed a small but efficient piece of software that will perform HTTP requests as well as receive, decode and display JSON objects at a desired rate. We have tested this software against a server setup akin to the one listed in the competition rules' appendix, and have made it robust enough to be re-configured for whatever server configuration is divulged to competition participants the day before the mission

Expected Task Performance

We expect to meet or exceed the "objective" set of accomplishment criteria set forth in the competition rules for each mission task outlined above. This expectation is based on the research that has led up to the development of our system, the testing we have performed at our home airfield, and the confidence in our team members' abilities to perform under pressure.

Programmatic Risks and Mitigation Methods

As with any R/C flight, a number of precautions must be taken in order to prevent potential damage to both people and property located in, at, or around the competition site. We have identified the following risk areas and will explain our mitigation methods for these areas below: loss of radio control, loss of telemetry link, loss of GPS signal, loss of live video feed, loss of image transfer network connection, and loss of power (aka total system failure).

In the event of loss of radio control, the GCS operator will first notify the safety pilot, then immediately notify all judges present. If the aircraft is in the middle of an autonomous mission, the loss of radio control alone will not trigger any change in behavior beyond a notification on the GCS software. If radio control is lost while the safety pilot is supposed to be in control of the aircraft, the flight controller will redirect the aircraft to the takeoff (home) point until radio control is restored. If the loss of radio control is coupled with the loss of telemetry, regardless of the flight's level of autonomy, the flight controller will redirect the aircraft to the takeoff (home) point until the telemetry link is restored.

In the event of loss of telemetry link, the flight controller will redirect the aircraft to the takeoff (home) point until the telemetry link is restored.

If the GPS signal is lost, the aircraft will climb at a slow, steady rate until the GPS signal is restored, then maintain current altitude until otherwise required by mission objectives.

If the live video feed link is lost, the aircraft will climb at a slow, steady rate until the live video feed link is restored, then maintain current altitude until otherwise required by mission objectives.

If the image transmission network link is lost, the aircraft's onboard computer will queue up the photographs being taken for transmission until the image transmission link is restored, at which point image transmission will resume. If the image transmission network link is unable to be restored, the images will be held on a microSD card for retrieval and will be uploaded to the GCS for analysis after the aircraft has landed.

In the event of a total system failure, the GCS operator will first alert the safety pilot that all systems have ceased to function, followed by a similar announcement to all judges. The safety pilot will then attempt to steer the aircraft away from any people or property nearby in the hopes that radio control may reach the aircraft intermittently. If necessary and possible, the GCS operator will also shout out to any persons that may be within the aircraft's crash trajectory or surrounding area.

Section 2: UAS Design

Description

Our aircraft has been designed with the mantra "as big as it needs to be in order to carry the required payload and maintain stable flight during mission tasks" in mind. We determined that an Aircraft Modelers' Research (AMR) Trainer 26 airframe would provide us with a large enough fuselage to incorporate our mission hardware while providing a stable aerial platform with which to utilize that hardware. The Trainer 26 was carefully built by members of the now-defunct Mizzou UAV Team using professional measuring and working tools with the operation being overseen by two senior team members with over 15 years of R/C experience between them.

The airframe is powered by a gasoline-powered DLE-40 two-stroke engine. In order to accommodate such a large engine, custom engine mounting plates made of carbon fiber were incorporated onto the base of the aircraft's nose. We decided not to include a cowl over the engine in our design since the aircraft is not designed to be flown in wet weather, and because the lack of a cowl offers more efficient cooling for the engine… it also looks downright gorgeous with the exposed engine and twin exhaust shafts; we have nicknamed our aircraft the "Sky Tractor" as a result.

We have chosen to use the APM 2.6 autopilot module as our flight controller due to its demonstrated ability to perform as expected, both on and off the field. While we are aware that the APM has been succeeded by the Pixhawk autopilot platform, we have not encountered any pressing reason to upgrade our system's autopilot; a wise man once said, "Don't mess with success", and we have no plans to mess with the core components of the system that helped us get ninth place in the mission category last year.

Our data links have been organized such that no communicating hardware will interfere with the communications of another piece of hardware. Our telemetry link will be established in the 900MHz frequency spectrum, our live video link on the 1.2GHz spectrum, R/C on the 2.4GHz spectrum, and image transmission will be done on the 5.8GHz spectrum. By selecting different spectrums for each mission-critical component of our system, we are drastically reducing the potential for signal noise on any one frequency spectrum.

The remainder of our payload consists of a Canon EOS 6D camera for completing the search area task, a Jetson TK-1 onboard computer for facilitating the transfer of photographs down to the ground station, an Ubiquti Rocket M5Ti access point for the air portion of the air-to-ground network, a DC to DC converter for powering the access point, a GPS compass, a telemetry radio, a Mobius ActionCam for live video, a 1.2GHz generic video transmitter to facilitate the live video feed, a GoPro Hero4 Black to record our mission for future viewing, and a GoPro Hero3 Silver as an emergency target image gathering apparatus.

Our GCS is made up of two high-powered laptop computers, one tablet, and a directional, beam-width antenna integrated with a router mounted on an antenna tracker. One of the laptops (GCS_1) will be dedicated to running the GCS software for the APM. The other laptop (GCS_2) will be connected directly to the router/antenna combo to receive the images transmitted from the UAS and process them using our target classification software. The tablet will be connected to a video receiver for displaying our live video feed and telemetry information during the mission.

The target data will be saved to a folder on GCS_2. This folder will be monitored for .jpg files by our classification software. As images are transmitted down to GCS_2, the software will read the image. It will then convert the image into the HSV color space. The image will then be scanned for objects no smaller than <pixel number here>. A 150x150 region of interest (RoI) will be cropped using the detected objects as a center point. These RoIs will make up the *candidate targets* of an image. These candidate targets will then be split into five "layers", so to speak; each pixel will be examined and placed into one of five categories. Each category will then be written to its own 150x150 image. The result will make up the *binary candidate targets* of an image. The binary candidate targets will each be analyzed to see if a clear, single shape can be detected. If so, we reason that the candidate target from which the binary candidate originated is, in fact, an image of a target. The candidate is then placed into a collection of *cropped targets*. These are then further analyzed for the characteristics of a competition target. If none of a candidate target's binary candidates contain a clear, single shape, we reason that the candidate does not contain a competition target and it is discarded.

Programming an autonomous mission and loading it into the autopilot will be done via a wired link for the initial mission. Additions, changes, or a reworked mission will be uploaded in flight using the 900MHz telemetry link. The emergent target waypoint followed by the landing approach and execution will be uploaded this way as well. The mission is laid out with a simple point and click method to add each waypoint on a Satellite image of the desired area with GPS data. Altitude, airspeed, rate of climb, tasks such as to activate a servo, and waypoint radius can then be set individually for each waypoint if desired, but will be pre-set at known optimal values for our airframe configuration to save time and optimize performance.

Autonomous Target Detection

As previously mentioned, we have developed a suite of software components that work together in order automatically find competition targets within an image and classify mission critical attributes about those targets. In order to test our software, we created our own targets using plywood and spray paint to create targets similar to ones used in the competition that were within the minimum and maximum sizes listed in the competition rules. We then deployed these targets randomly throughout our local airfield and sent the Sky Tractor up to take some pictures. The images generated by these test runs has made up the majority of our software tests. We have also experimented with superimposing digital representations of competition targets onto aerial photographs obtained with Google Earth in order to provide a more difficult challenge for our software to overcome. While this testing method is less realistic than our home field simulation, we feel that ensuring that the software can handle a wide variety of background and target combinations will lead to a greater chance of successful and quick autonomous detection and classification in the competition.

A (somewhat poorly cropped) image of our aircraft

Section 3: Test and Evaluation Results

Mission Task Performance

Our first tests were done to ensure that our UAS would be able to complete the two primary mission tasks, autonomous flight and traversing the search area. Our aircraft performed flawlessly for the autonomous flight task, and we re-established that confidence in our system by programming a simple takeoff, an approximately three-minute flight with four waypoints, and landing. Our aircraft performed this task with no problems as expected.

We then placed the camera in the aircraft and proceeded to execute the same autonomous mission outlined above, this time with our camera trigger script, air-to-ground network, and target recognition & classification software active. Prior to mission execution, we spread our test targets out randomly in the field. Our image transfer network connection broke once due to the antenna transmitter getting stuck, but the remainder of the mission was carried out by manually aiming the beam antenna at the aircraft. The connection was restored after forty-two seconds of connection downtime and image transmission resumed one minute and sixteen seconds after the initial disconnection (thirty-four seconds after the connection was re-established). This caused us to not have all the images by the time the plane landed, but we allowed the plane to sit on the runway until the images finished transferring. We do not expect this to be an issue at the competition as the additional processing time required beyond landing the plane did not exceed ten minutes. We are also replacing the antenna tracker's moving parts to further reduce our likelihood of an image transmission network disconnection.

Next, we tested to make sure that the camera would properly take photographs, send them to the on-board computer, and then be transmitted to the ground station. Picture taking is done through a script running in the background on the Sky Tractor's on-board computer; the camera is set to take a new photograph approximately every three seconds. While this does result in a large number of images, our software has demonstrated its ability to analyze a 5,000 x 4,000 pixel image for targets in under ten seconds, and can determine whether or not an actual target is present in the photograph in under 30 seconds. We simulated a break in the wireless connection by standing directly in front of the beam antenna until the connection broke, then stepping away to resume transmission. There was a longer period between re-establishing the connection and images resuming transmission than we expected, but other than that the wireless network and related processes are working as planned. Our software programmers are looking into why the images take longer than normal to resume transmission and this issue will hopefully be solved by competition time, but we are not concerned about this issue impacting our competition performance if it is not solved.

The test outlined above was also used for testing our projected performance on the ADLC and Actionable Intelligence tasks. As noted earlier in the paper, we are currently experiencing a >50% FAR, but are meeting all other objective-level elements in the mission task. Our coding team is working hard on improving the algorithms that determine whether or not a candidate actually contains a target. We are excited to note that, despite the targets being laid in some rather unkempt areas of the field, our software was able to detect all of the targets despite being partially occluded by brush.

To test for the off-axis task, we set up another autonomous mission where the plane was instructed to bank heavily while flying as close to the edge of a given boundary as possible. A target was placed approximately 300 feet away. Subsequent image analysis on the photos obtained while performing the banking maneuver were acceptable; our software was able to determine the color of the target, the color of the character, and the character itself. Our confidence in the completion of this task is only diminished by our inability to test for this task where the off-axis target is 500 feet away from the boundary (due to home field dimensions).

For the emergent target task, the two senior members of the team created a target without the knowledge of the other members. We then placed this target on the field mid-mission without showing the other team members the target (we were carrying out a dry run for the off-axis test). We then instructed one of those team members to add a new waypoint to have the plane fly over the "mystery" target's location. The non-senior team members were then asked to look at the photographs taken over the "mystery" target and identify it. The target in question was a six foot by three foot poster-sized image of a man swimming. The non-senior team members were able to identify this image. We would have liked to have one of our members dressed up embarrassingly in a dress and hidden out in the field prior to the other team members arriving, but our home field's rules would not allow for that; we had to negotiate to allow a team member to go out on the field while an aircraft was in the air. Nonetheless, we are confident that we will be able to capture an image of and describe the emergent target.

We were not able to directly test for the SRIC task due to lack of power at our home field. However, we did test the components of this task individually. For the flight component, we found the slowest rate at which we could circle an object from a given distance. We found that we were able to circle an object from any altitude within the competition limits at a radius of up to approximately 75 feet. On the ground back at the lab, we discovered that we would be able to connect to a beam width transmission from this distance, but not with much room for margin of error. We are working on developing a contingency plan in case of a disconnect, or perhaps trying to perform the tasks in a series of passes.

Our air-drop test was done next. We placed one of the targets out on the field and sent the plane up. The autopilot was then instructed to activate the servo holding the bit of string on the egg at a given location such that it would hit the target. We then programmed the aircraft to fly over the target a total of four times. The first three passes over the target, our safety pilot did not engage the air-drop safety; as a result, the autopilot was not able to cause the servo to move. On the fourth pass, the safety pilot armed the air-drop safety and the flight controller was able to successfully move the servo such that the string snapped and the egg was sent below. We did not strike the target due to wind gusts that had not been taken into consideration, and we have not had the opportunity to test for this more than once; we plan on dedicating a number of days for learning how the egg drops at various speeds and wind conditions.

Finally, for the Interoperability Task, we have tested our software against a simulated remote clock and GPS system. We were able to achieve the desired 10 Hz refresh rate and were also able to successfully limit the refresh rate to 14 Hz, leaving a small margin for fluctuations in timing.

Section 4: Safety Considerations and Approach

With two returning competitors and two first time competitors, we made the decision to train all team members on all aspects of handling the competition tasks at the flight line. The mission itself can be a very intense experience, and we thought it best that all team members be qualified to fulfill any position throughout the competition in the event of an emergency. Each team member has more experience in one aspect of flight line operations than the others, but each team member is qualified to "fill in" for another should the occasion arise.

We have taken care to dedicate a different frequency spectrum to each data link between the UAS and the ground station. This significantly reduces the chances of critical systems failure due to a lack of communication through jamming or interference resulting in injury to person or property.

We have prepared a number of failsafes and contingency plans as outlined in the "Programmatic Risks and Mitigations" section of this paper. In doing so, we feel that we are doing our part to protect the people and property that make this competition possible, as well as our own assets that we have invested our time, money and passion into.

Lastly, we have ensured that we meet or exceed all of the safety requirements outlined in the competition rules.

Conclusion

In conclusion, based on the information stated in this journal paper, we are confident that our performance this year will be a successful one. We would like to thank the organizers of this competition for continuing to host an event that fosters such a unique and competitive environment for UAS enthusiasts in higher education across the world. We are excited to spend another weekend making new friends and learning about other schools' teams and design methodologies.