2015 AUVSI STUDENT UAS COMPETITION

POLITEHNICA University of Bucharest

Phoenix Team Journal Paper

ABSTRACT

This paper provides a summary of the POLITEHNICA University's UAS, *SkyEye*, designed to meet the objectives of AUVSI student UAS competition. *SkyEye* is a custom built quad copter controlled by the ArduPilotMega, an open-source autopilot. Capable of following dynamically changing waypoints, *SkyEye* provides real time reconnaissance to an Imagery terminal on ground using a gimbaled stabilized point and shoot camera. The transmission of captured images takes place on a 2.4GHz secured wireless link. The received images are then processed for actionable intelligence. Modular in design, *SkyEye* can be brought to flying state in less than 10 minutes. Safety being of paramount importance in all aspects of UAS operations, *SkyEye* can be controlled by its Ground Control Station over a 915 MHz radio link as a Remotely Piloted Vehicle (RPV) and also by a 2.4 GHz Radio transmitter remote under full manual control.

This paper gives a detailed description of *SkyEye*'s system along with the design rationale including the team's objectives in building the system. The paper also describes the UAS flight operations and concludes with a description of the testing that has been performed to show that the mission can be completed safely and successfully.

Table of Contents

1 Description of the systems engineering approach

1.1 Mission requirements analysis

The AUVSI student UAS competition simulates a real life mission that has to:

- provide intelligence, surveillance and reconnaissance (ISR) support using Unmanned Aircraft Systems (UAS) which has to:
  - perform autonomously,
  - must comply with Special Instructions (SPINS) for departure and arrival procedures,
  - remain within assigned airspace,
  - search an area for items of interest,
  - conduct point reconnaissance.
- be capable of receiving vital messages from Simulated Remote Information Center (SRIC)
- provide thermal imaging to locate and track firefighter's positions.
- accurately deliver retardant or water where directed.

Based on the CONOP's, a Key Performance Parameters (KPP) chart was prepared, that was used by the team to identify goals on which further work was to be done once threshold were complete:

	Parameter	Threshold	Objective
Primary	Take off , flight and landing	Achieve controlled takeoff, flight and landing.	Achieve controlled autonomous takeoff, flight and landing.
	Waypoint navigation (each waypoint)	Capture waypoint in sequence.	Capture waypoint autonomously in sequence.
	Localization (each target)	Determine target location within 100 ft.	Determine target location within 50 ft.
	Classification (each target)	Identify any two target characteristics.	Identify all five target characteristics.
	Autonomous Search	Fly the search area.	Fly the search area autonomously
Secondary	SRIC task	Download the secret SRIC message.	Download the team's SRIC message and perform the task defined in the message.
	Air-drop task (release)	Manual release	Autonomous release
	Drop Accuracy	<100 ft. from bulls-eye.	<50 ft. from bulls-eye.
Restrictions	Mission Time	Within 40 minutes	20 minutes
	Aircraft	Max. GTOW 55 lb	Minimum GTOW
	Control	Manual (at all times)	Manual plus, through GS as RPV for added redundancy.
	Communications	Different radio bridges for manual control and GCS	Different networks for each of imagery, manual and auto control
	Safety	Failsafe: Land	"Return Home" on link failure and Autonomous landing.
	Cost	Minimum for primary tasks	Minimum for all tasks

1.2 Design rationale

The team followed a structured Systems Engineering approach to face the 2015 SUAS challenge. The design process was divided in 3 major phases:

1. Analysis 2. Preliminary Design 3. Systems Integration and Testing.

In Analysis phase, the team thoroughly studied the KPP charts prepared to direct the improvements in each sub-system. The team allocated their resources and time on the basis of these KPP charts.

The Preliminary Design phase involved subsystems integration, after which extensive laboratory testing and field testing was done. The components were required to perform reliably without any failure. This practice proved beneficial later as it facilitated our systems integration effort, due to dependability of our subsystem modules.

The final phase, Systems Integration and Testing, involved putting together all the subsystem modules on one platform, which were then tested in flight. The behavior and performance of the complete system was observed and necessary alterations were made.

Design rationales have been conducted in the Preliminary Design through Figure of Merit (FoM) tables for key features as follows:

- Image capturing subsystem
- Image processing subsystem
- Aerial platform
- Communication system
- Ground control

Different solutions were compared against four criteria:

- Mission requirements
- Safety
- Weight
- Cost

The imagery system was critical in the design process. A camera with a narrow angle lens was chosen, for more detail on the target. Mounted on a two axis gimbal, the camera is able of being pointed in any direction addressed by the payload operator. All images gathered by the imagery system can be transmitted in real time back to the ground station for evaluation, but most image processing is done onboard the airframe, in background, for time noncritical operations.

The aerial platform was designed to be capable of vertical takeoff and landing (VTOL) with the ability to accomplish autonomous flight, including autonomous takeoff and landing, while operating an imagery payload. The use of a rotor-wing aircraft was decided early in the design process. There are several characteristics of a VTOL aircraft that were thought to be beneficial during competition. A feature rotor-wing aircraft boasts over fixed-wing is maneuverability; this is valuable in reconnaissance type missions since the aircraft is able to move three dimensionally at low speeds and low altitudes. Above all other advantages, a VTOL aircraft has the ability to hover. When surveying targets with the intent of gathering actionable intelligence, hovering over an object can supply a much more stable look at a target rather than passing over without stopping.

While building the aircraft, the competition objectives and parameters were addressed during design. The method of autonomy was addressed by implementing a commercially available autopilot. Simple navigation tasks are accomplished by plotting waypoints the aircraft will follow. Re-tasking is as simple as adding additional waypoints for the aircraft to track to.

The system was designed to be suitable for surveillance purposes which imply presentation of position and other data transferred during flight. An important goal was the use of commercial off-the-shelf components, making it a low cost option.

1.3 Expected task performance

Performance of the system was streamlined by integrating the imaging system with the autopilot system. The autopilot chosen works seamlessly with the imaging software that powers the camera setup. By integrating these two systems, the issue of complexity by adding extra hardware for camera control and stabilization was eliminated. Advantages that came with this included reduction of weight added by payload, reduced chance of component failure, and overall cost of the system is reduced.

With the VTOL ability of the aircraft, autonomous takeoffs and landings are an unassuming task carried out by the aircraft slowing down, and making a gradual vertical decent onto the landing platform.

In the design process of the aircraft, the competition rules and objectives were constantly kept in mind. The rules list a number of parameter thresholds and objectives, and with the ongoing flight test, the aircraft consistently shows that it can operate at the threshold parameters and most of the objective parameters. It is anticipated that the majority of the 40 minutes will have to be used for the mission. The relatively small endurance of the aircraft means at least one change in batteries to complete the mission. Target imagery and location continue to be tested, and with more time spent, it is anticipated that the objective parameters will be met in time for competition.

It was found that the area to be scanned is roughly 250000m^2 (2.7*10^{6}ft^2). By field testing of the chosen camera, altitude for reliable target recognition, while keeping the amount of pictures taken reasonably, was found to be 70m (230ft). Under these assumptions, a number of 40 pictures will cover the search area, with a resolution of 2cm/pixel and a ground footprint of 97x72m. The flight time for this survey is 19min. Adding 5min for waypoint navigation before entering the search area and 10min for second path at lower altitude above the identified targets, flight time reaches 34min. This leaves very short time out from the 40min limit, being unsafe to try other secondary tasks.

If more than one flight will be scheduled for our team, other secondary task will be attempted, i.e. SRIC task, Emergent target task and Drop test task (for which the payload will have to changed)

1.4 Programmatic risks and mitigation methods

As the rules don’t state the relative importance of the tasks (the scores) it is team guess that most valuable secondary tasks are the ones related to image recognition. The risk is to be wrong with this guess. If this is the case, the team is prepared to revert to manual target recognition, taking only one path over the search area. This will give team the opportunity to try other secondary tasks within the 40 min window.

Also, search area size is to be guessed from little hints given. The risk of wrong guessing the search area size can be mitigating by raising the altitude while searching, for reducing the number of necessary pictures to be taken, and thus reducing the time for covering the entire area to meet the threshold requirements.

The risk of not meeting the design weight of the aircraft will lead to a reduction in endurance which can be mitigated by replacing propulsion battery within the 40 min. window but with the penalty in time for doing this. This will be alleviated by going for manual target recognition and attempting other secondary task after battery replacement.

2 Descriptions of the UAS design

2.1 Design descriptions of the aircraft, autopilot system, data link, payloads, ground control station, data processing, and mission planning

Our system is meant to provide real-time recognition of objects on the ground from a UAV, while requiring only a low bandwidth radio link between the aircraft and the ground station. The solution is for the aircraft to do an initial pass of image recognition to find "interesting" objects on the ground, and then to show small thumbnails of those objects on the ground station, overlaid on a satellite map of the area. The operator can then select which of these thumbnails to look at more closely, which will lead to an in-flight mission reconfiguration (new waypoints at a lower altitude). This will allow bringing down a full high resolution image around that object from the aircraft over the Wi-Fi link.

When using this method the operator gets a complete overview of the search area, and can quickly focus on areas of interest. Included in the system is a geo-referencing system that uses the MAVLink telemetry stream along with image timestamps to work out the geographic location (latitude/longitude/altitude/heading) of any region of interest in the image.

2.1.1 Air Vehicle

In order to perform the competition challenge, Phoenix Team has selected a quadcopter as an aerial vehicle because of its strong adaptive properties and its ability of carrying extra load. We choose it due to agile maneuverability and advantage of mechanical simplicity. On the other hand it`s capable of hovering and it`s more stable while taking pictures. Due to high resolution pictures for targets extraction, quadcopter has good vibration mitigation and strong enough frame.

“Sky Eye” is a quadcopter made entirely by carbon fiber with arms diameter of 16 mm and its distance between two opposite tubes of 650 mm. The tubes are empty inside for reasons of weight reduction. The command center spitted into three main levels, houses the whole system. On the upper level are set the image processing unit, ODROID-U3, Pixhawk Autopilot, GPS with magnetometer, Wi-Fi, telemetry, power distribution and SSC with engines. The second level, manufactured at CNC (Computer Numerical Control) by duralumin alloy, supports just the battery packs. On the lower level is fixed the gimbal and QX 10 camera kit . Gimbal is able to point the Sonny QX10 camera ± 45 degrees from nadir in the roll and pitch direction. The Gimbal is commanded by two mini digital metal gears servo. It is integral manufactured with 0.059 in (1.5 mm) thickness glass textolite.

Weights of main parts are shown bellow.

PART	WEIGH (lbs)
Scout total	5.28
Frame	1.36
Motors	0.86
Battery	1.78
Camera System	0.65
Autopilot	0.04
ODROID-U3	0.22
Gimbal	0.37

2.1.2 Propulsion System

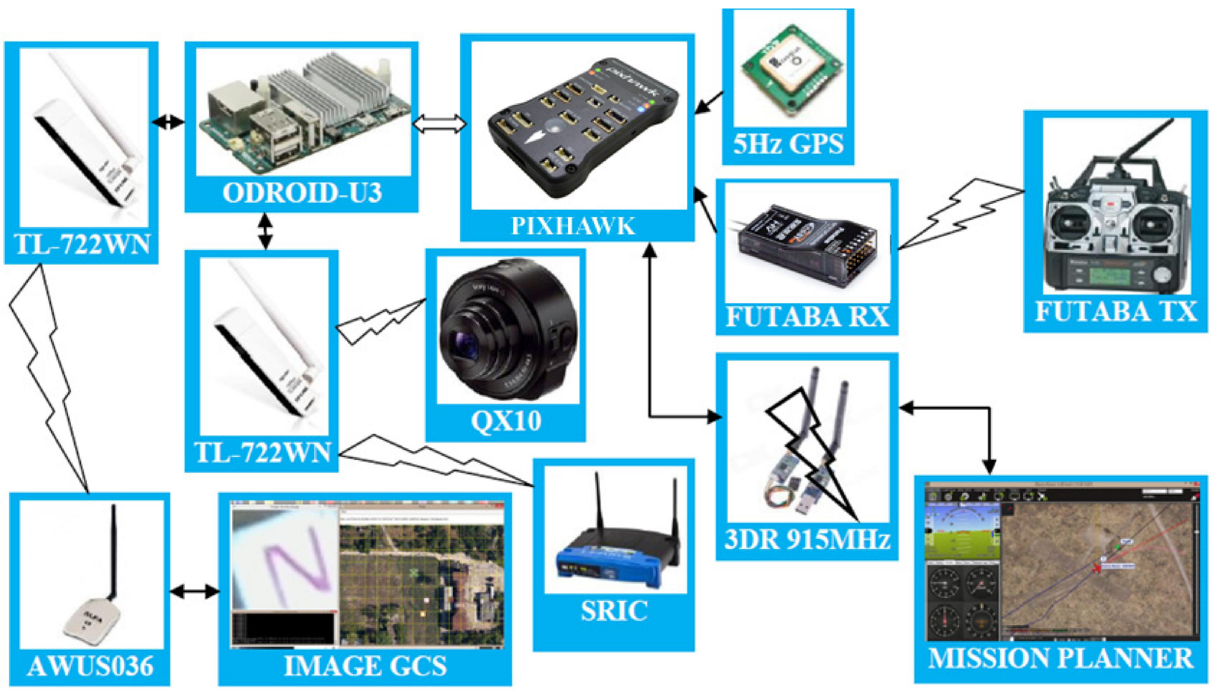
The propulsion system has several fundamental requirements: provide enough power to complete the mission’s tasks, maximize the cruise speed for area survey all while maintaining a low system weight. The selections of motors were highly important as to the overall performance of the quadcopter. The choice was primarily based on the efficiency of the motor, as well as the weight and electrical output of the system. Through the medium of EPROP–calc program, assembly composed of motor-propeller and propulsion pack was computed for 40 min. endurance. Taking into account current draw, power ratings, size factor and weight a 48-22-490 kV electric motor combined with 16x5.5 carbon propeller performed the challenge due to algorithm sampler. Therefore, 11000 mAh Lithium Polymer cell was chosen to maximize power output while minimizing weight. The cell’s ability to draw high current for a large time interval, as well as maintain a nominally constant output voltage, was an important design constraint. The battery chemistry provides a high-capacity density and a very repeatable charge and discharge cycle and gives a small voltage drop under high amperage loads. Assuming 90% system efficiency it was determined that 1 cell battery configuration was well-suited for the quadcopter.

2.1.3 Manufacturing

The manufacturing process included construction of both experimental devices (the main quadcopter and its backup). The main parameters considered when deciding upon construction were:

- Manufacturability – This includes ease of manufacturing by unqualified personnel
- Precision- Precise construction could improve quadcopter performance. Creative solutions had to be invented in order to save construction time and maintain high manufacturing precision.
- Weight- Ability to manufacture weight efficient parts with enough strength
- Commercial availability and price – The commercial availability of parts that the team decided to order like carbon tubes, textolite, and duralumin alloy. The parts delivered had to be bought at a reasonable price and supplied in proper time.
- Assembly time – The parts constructed had to be quick to assemble in the field, at the competition.

2.1.4 System Architecture



The system consists of Pixhawk Autopilot, ODROID-U3, QX10 and GCS. Pixhawk is a high performance autopilot-on-module suitable for our quadcopter, which means it supports both piloted and fully autonomous flight, including GPS waypoints, camera control, auto takeoff and landing. ODROID-U3 is the image processing unit, a powerful credit card size Linux computer with 1.7 GHz Quad-Core processor, 2GByte RAM, 8GB SSD, 100 Mbps Ethernet LAN and High Speed USB2. Image capture is done through a Wi-Fi enabled SONY QX10 camera. The GCS is composed of 2 PC. The first one supports the command, control and monitoring of the UAV flight in real time, providing the flight management of the UAV. The second one receives the processed target images from ODROID-U3, with EXIF positioning data embedded in the images, displays them on a moving map, in real time, and does the feature recognition on them.

Images processing, such as segmentation and recognition, are done with powerful numerical routines from the OpenCV library (Open Source Computer Vision Library). With the aid of background subtraction, like HSV, mean-shift segmentation and Delaunay filtering, the targets from on-board images are recognized. Crop images of the targets along with EXIF embedded geo-tagging information are transferred to the Ground Control Station for shape recognition and OCR.

The recognition algorithm works in two stages, a first stage that finds anything unusual in the image, then a second stage that converts that region to HSV color space and applies some simple heuristics to score the region. User is able to easily plug-in different scoring systems to suit their own image search task.

The cropped images are subjected to numerical search algorithms to extract contours features, which are then classified by their geometric properties (number of corner, convexity of the hull, etc.) in order to recognize their shape and background color. Inner contours are then passed to a neural network system trained to do OCR in order to find the letter in the target. Finally, earth geode routines are used to determine the orientation (heading) of the target.

2.1.5 Autopilot

As a full UAV autopilot, we choose an APM, in order to support fully autonomous flight, including hundreds of GPS waypoints, camera control and auto takeoff and landing. APM offers: three axis camera control and stabilization, shutter control, live video link with programmable on-screen-display, data transceivers allow real-time telemetry and control between our ground station computer and APM, including joystick control options, full data logging provides comprehensive post mission analysis, with graphing and Google Earth mapping tools.

2.1.6 Mission Planner (Primary GCS)

Mission Planner is an open-source ground station (GCS) application for MAVlink based autopilots including APM and PX4/Pixhawk that can be run on Windows, Mac OSX, and Linux. Mission Planner allows you to configure a plane, copter, or rover to work with the autopilot to become an autonomous vehicle. Use Mission Planner to calibrate and configure the autopilot, plan and save missions, and view live data in flight.

The above is the main Ground Station view of the Mission Planner, showing the Heads-up Display (HUD). Once you have connected via MAVLink over USB or wireless telemetry the dials and position on this screen will display the telemetry sent by APM.

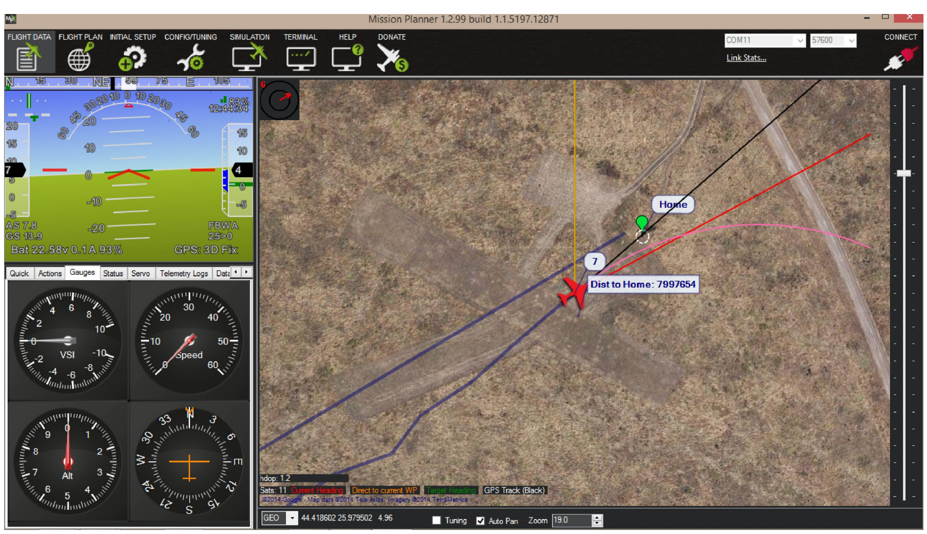


Fig. 2.1. Mission Planner

In Mission Planner we can create missions using the easy point-and-click Waypoint Editor. One of the most commonly-used features in pro UAVs is point-and-click mission control in real time. Rather than just pre-planned missions or manually flying the UAV, operators can just click on a map and say “go here now”.

Mission Planner is used as primary GCS for controlling autonomous flight, editing waypoints for accessing search area and building waypoints path and commands for area survey. An example is provided in figure below.

Fig. 2.2. Waypoints editor

2.1.7 MAVProxy (Secondary GCS)

MAVProxy is a fully-functioning GCS for UAV's. The intent is for a minimalist, portable and extendable GCS for any UAV supporting the MAVLink protocol (such as the ArduPilotMega).

- It is a command-line, console based app. There are plug-ins included in MAVProxy to provide a basic GUI. For a GUI that works on tablets.
- It is written in 100% Python.
- It is open source.
- It's portable; it should run on any POSIX OS with python, pyserial, and select() function calls, which means Linux, OS X, Windows, and others.
- It supports loadable modules, and has modules to support console/s, moving maps, joysticks, antenna trackers, etc.

Due to its modularity, only required MAVProxy modules are used in special designed software for this application.

Secondary GCS is used to monitor target recognition, during flight. It uses the map module, flight path module and geo-tagging module. As flight progresses, successive UAV’s positions are dotted on the map and, when a possible target has been found, target thumb is geo-tagged on the map and displayed in a separate window (

Fig. 2.3.). The operator has the possibility to accept or reject target for further processing.

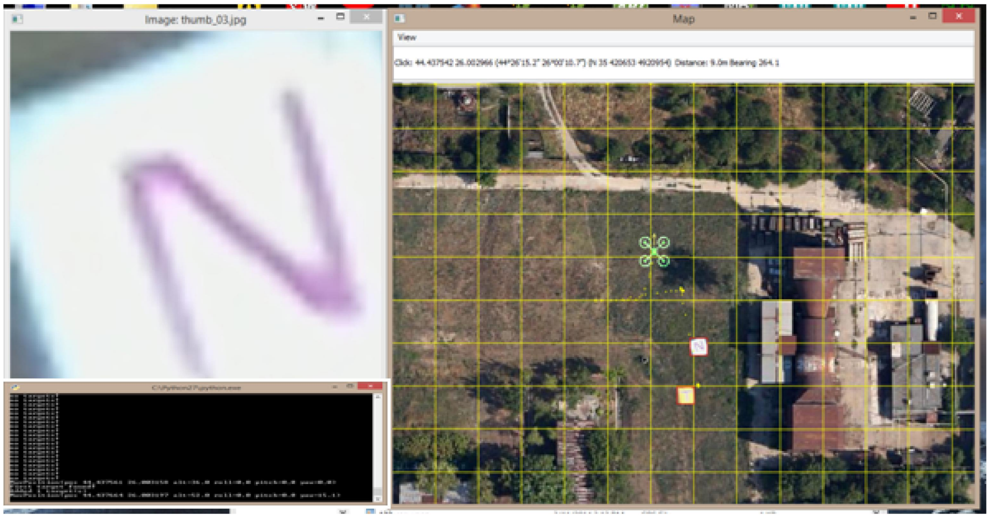


Fig. 2.3. Secondary GCS

2.2 Target types for autonomous detection

2.2.1 Target Recognition and Extraction

Once entered the search area, the UAV follows a predefined path and takes pictures in certain points as to cover all area with 20% overlapping pictures (Fig. 2.4.). Due to altitude limits imposed (min. 100ft., max. 750ft.), picture resolution has to be high (18MP) in order to have enough resolution of the extracted targets, whose dimensions are between 0.6 to 2.4m (2 to 8ft).

As seen from Fig. 2.4. the search area has about 227000m^2, and, from 150m altitude, it takes 35 pictures for covering the entire area. Each picture cover a 153x114m rectangle leading to 6cm/pixel resolution. Target resolution will be, under these circumstances, between 10 to 240 pixels.

In order to simplify the recognition algorithm, camera stabilization is used, implemented in the Autopilot software. The software assures vertical stabilization of the optical axis.

Due to high resolution of pictures, sending them over Wi-Fi to the secondary GCS will take too long. This is why target extraction from the original pictures has to be done onboard UAV, and this is where the ODROID-U3 proves it's utility. Due to its processing power, it is capable to process one picture and extract targets in less than 10s. Next, only thumbnails of target found are sent over the downlink for further image recognition. Thumbnails transfer time is less than 5s over a 54mB Wi-Fi link.

Target recognition and extraction is done by means of background subtraction algorithm.

First, the 18MP image acquired by the camera is stored into an $4920 \times 3264 \times 3$ array. Each array element contains a pixel's color information (Red, Green or Blue channels). Next, color space conversion to Hue Saturation Luminance (HSL) of the image is performed.

Using histogram back projection algorithm region of interest (targets) in the image are extracted to thumbnails. This algorithm is used for image segmentation or finding objects of interest in an image. In simple words, it creates an image of the same size (but single channel) as that of our input image, where each pixel corresponds to the probability of that pixel belonging to our object. We create a histogram of an image containing our object of interest. The object should fill the image as far as possible for better results. And a color histogram is preferred over grayscale histogram, because color of the object is more better way to define the object than its grayscale intensity. We then “back-project” this histogram over our test image where we need to find the object, i.e., we calculate the probability of every pixel belonging to the ground and show it. The resulting output on proper tresholding gives us the ground alone, which can be subtracted from the image to obtain targets (Fig. 2.5).

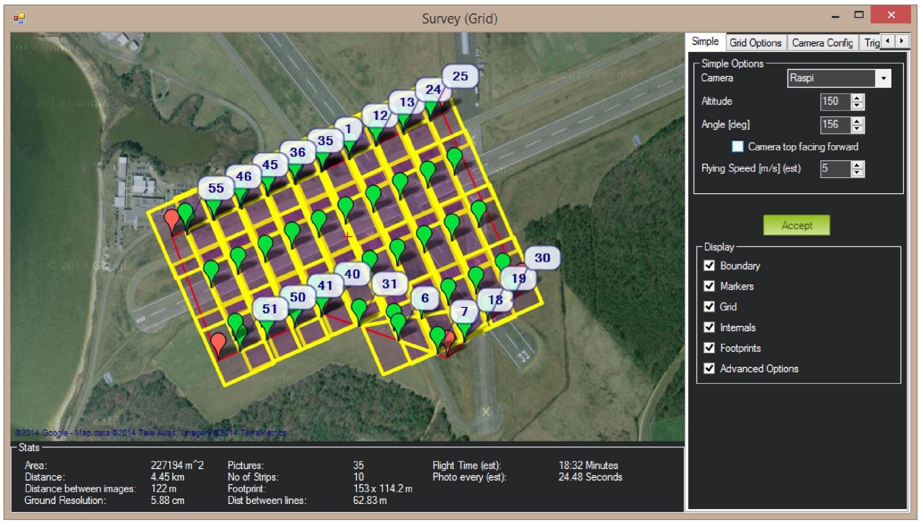


Fig. 2.4. Search area pictures

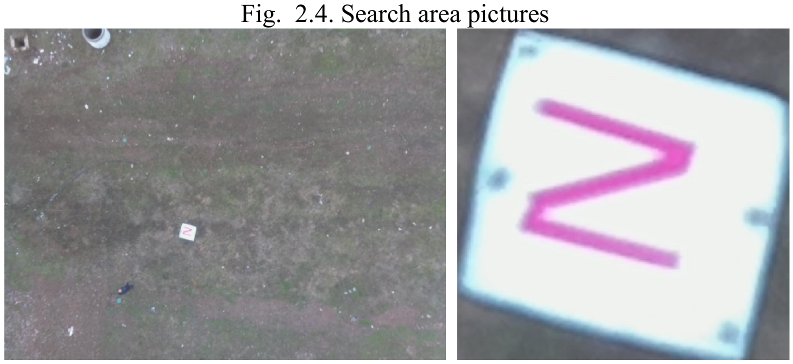


Fig. 2.5. Target (thumbnails) extraction

2.2.2 Finding contours

Contours can be explained simply as a curve joining all the continuous points (along the boundary), having same color or intensity. The contours are a useful tool for shape analysis and object detection and recognition.

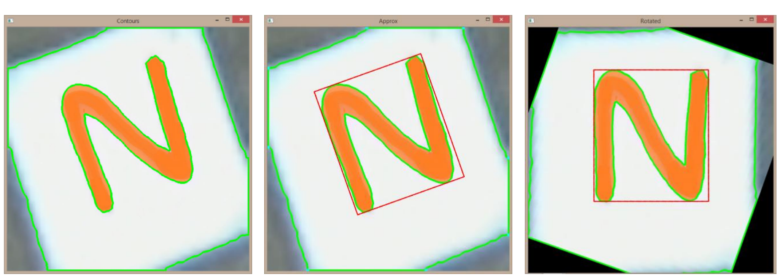


Fig. 2.6. Contours extraction

Image moments assisted us with description of objects after segmentation, estimation of mass center, or object area. Simple properties of the image which are found via image moments include area, total intensity, its centroid and information about its orientation.

The function computes moments, up to the 3rd order, of a vector shape or a rasterized shape. The results are returned in the structure Moments.

As an increase of accuracy, *Contour Approximation Method* helps us to remove all redundant points and compresses the contour. The functions *approxPolyDP* approximate a curve or a polygon with another curve/polygon with fewer vertexes so that the distance between them is less or equal to the specified precision. We approximate the basic contour shape to another shape with less number of vertices, by the aid of *Douglas-Peucker* algorithm.

2.2.3 Target Shape Detection

Another problem to solve is the shape of the target. This is done using feature detection algorithms.

First, corners of the found contours in the thumbnails are determined. After that, eigenvectors and eigenvalues are found. The output of the function can be used for robust edge or corner detection. The function *precornerdetect()* calculates the complex spatial derivative-based function of the source image. The corners can be found as local maximums of the function. The function *cornersubpix()* iterates to find the sub-pixel accurate location of corners or radial saddle points. Sub-pixel accurate corner locator is based on the observation that every vector from the center q to a point p located within a neighborhood of q is orthogonal to the image gradient at p subject to image and measurement noise. The algorithm sets the center of the neighborhood window at this new center q and then iterates until the center stays within a set threshold.

2.2.3.1 Circle detection

The idea of circle detection algorithm is based on the assumption, that all circle's edge points have the same distance to its center, which equals to circle's radius. Of course doing different image processing tasks it may happen that circles may have some distortion of their shape, so some edge pixels may be closer or further to circle's center. But this variation in distance to center should not be very big and should reside in certain limits. If it is too big, than most probably the object has other than circle shape.

2.2.3.2 Detection of quadrilaterals

Detection of quadrilaterals and triangles has pretty much the same idea - we are checking mean distance between provided shape's edge pixels and the edge of estimated quadrilateral/triangle. The only difference here is the way how to estimate parameters of the shape we want to recognize and how to check distance to the estimated shape.

For a given shape we can make an assumption that it is a quadrilateral, find its four corners and then using similar method as we've used for circles we can check if our assumption is correct or not As we can see on the image in Fig. 2.7., we may have different objects and quadrilateral. Finder provides different results for them. For shapes which really look like quadrilateral, the quadrilateral finder is able to find their corners more or less correctly (problem may occur in the case if object has rounded corners). But for other types of objects (circles, starts, etc.), quadrilateral finder does not provide any good result. And this is correct, since this routine supposes the given shape is really quadrilateral.

Now, when we made the assumption that a subjected object is quadrilateral and we got its corner, we just need to see how good is the fit of the object into the quadrilateral with those found corner - we need to make sure there are no edge points of the given shape which are too far away from the edge of the assumed quadrilateral. And here we'll apply the same algorithm as we used for circle checking. The only difference is the way of calculating distance between point and the closest quadrilateral' edge. The code goes through the list of all provided shape's edge points and finds distance between them and the assumed shape. Since we deal here with quadrilateral, the distance between point and the assumed quadrilateral is calculated as a distance to the closest quadrilateral's side. In the end of the above calculations we receive the same as before - mean distance between given edge points and the edge of the assumed quadrilateral. If the mean distance is not greater than a certain limit, then the shape seems to be a quadrilateral; otherwise it is not.

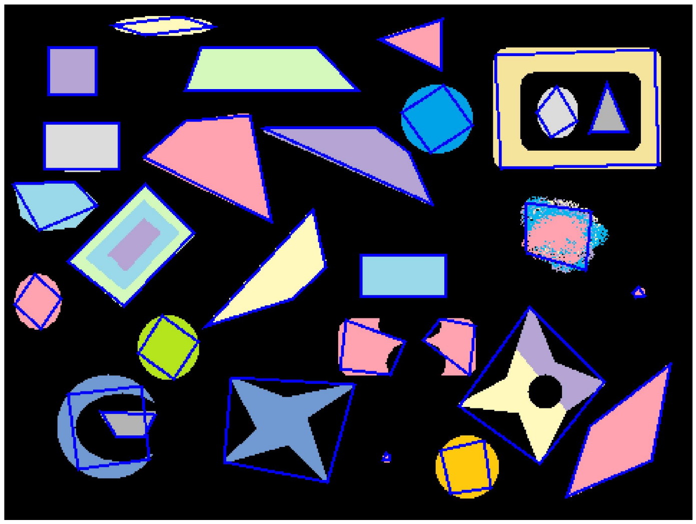


Fig. 2.7. Simple shape recognition

2.2.3.3 Detection of triangles

Idea of detecting triangles is almost the same like the idea of detecting quadrilaterals - just need to find 3 corners of the supposed triangle and then check the fitting of specified edge points into the assumed triangle. If the function returns only 3 corners, and it cannot find the 4th one, it is the case of triangles.

2.2.3.4 Rectangles, squares, equilateral triangles, etc.

Once we made a check for quadrilateral shape and got its 4 corners, we may do some further checks to detect sub-type of the quadrilateral: trapezoid, parallelogram, rectangle, rhombus or square. These checks are based on checking angles between opposite/adjacent sides and their length. First we check two opposite sides - if they are parallel, then we get at least trapezoid. Then we check two other sides - if they are parallel two, then we have parallelogram. If two adjacent sides are perpendicular, then we got rectangle. The last check is to compare length of two adjacent sides - if they are equal, then parallelogram becomes rhombus and rectangle becomes square. And of course we need to add small possible error, so if angle between lines equals to 2 degrees, they are still treated as parallel.

2.2.4 Optical Character Recognition

After all this algorithms applied on the contour detection, inner contours are then passed to a neural network system trained to do Optical Character Recognition (OCR). The goal of Optical Character Recognition (OCR) is to classify optical patterns (often contained in a digital image) corresponding to alphanumeric or other characters. The process of OCR involves several steps including: gets the center of mass, find the longest radius, gets the track-step, divide the object into tracks using the track step ,gets the sector step, divide those virtual tracks into equal sectors using the sector step, find relations between adjacent pixels, putting all the feature together. Tesseract open software is used for character recognition.

2.3 Target types supported by autonomous detection (if utilized)

As seen from the detailed design section above the ADLC system does not rely on templates in recognizing targets characteristics, except OCR. Due to this, team is confident that the target recognition algorithm will be capable of treating autonomously any type of target within the limits stated by the rules (2 to 8ft, basic geometric shape, 50 to 90% alphanumeric character).

Yet, one possible problem might be the description of target and character colors. While system can find mean color in numeric BGR format, a standard literal classification of colors is not available. Although, team has found over 800 colors names over the internet, the possibilities being much more (1.6billion colors), we have to rely on "nearest best match" to state the color, which can lead to mistakes, when done automatically.

2.4 Mission tasks being attempted

As the SUAS committee's intent to supply more tasks than can be completed in the available mission time, and as team has found, analyzing KPP, mission tasks being attempted have to be prioritized.

Primary tasks are of paramount importance in achieving a good score. So they are mandatory tasks from team's point of view. The whole design is focused in fulfilling the requirements of these tasks: AUTONOMOUS FLIGHT TASK and SEARCH AREA TASK.

As stated before, depending on the evolution of the contest, secondary tasks will be attempted in a priority order chosen to increase the team score, without jeopardizing the fulfillment of primary tasks.

ADLC and ACTIONABLE INTELLIGENCE tasks will be attempted if the assumptions in §1.4 are correct. If not, and attempting these task will lead to the risk of going over endurance of the aircraft, these task will not be attempted.

IR SEARCH task will not be attempted because a suitable sensor was not available to our team due to financial restrictions.

OFF-AXIS TARGET, EMERGENT TARGET, and AIR-DROP task will only be attempted in the event flight time will be available to team, because they require either software or hardware reconfiguration of the aircraft (video streaming, imagery payload replacement with drop device)

SRIC, INTEROPERABILITY and SDA tasks will only be attempted if the primary tasks are fulfilled and enough mission time is left for safely undergoing this task, because it does not require reconfiguration of the air vehicle, except Wi-Fi network software configuration which can be easily done through simple command line connection to the on-board ODROID computer.

3 Test and evaluation results

3.1 Mission task performance

3.1.1 Flight testing

As of May 24th, a total of 345 minutes of flight including 26 autonomous takeoffs and landing cycles were accomplished by the team. Throughout the flight testing "SkyEye" has shown the ability to take off in winds up to 15m/s (30kts), maintain a forward velocity of 5 to 20m/s (10 to 39kts), and demonstrated a programmable climb and a descent rate between 0.5 to 5m/s (1 to 10kts). To complete autonomous take off and landings a Sonar sensor was installed to assist the autopilot, from the first test flight autonomous takeoffs and landing have been carried out with less than 5 feet of error from the designated landing area. When testing the waypoint tracking feature of the APM, a qualified external pilot is on standby in case the aircraft performs undesirably. During test flights the aircraft is commanded through the GCS to fly to assigned waypoints, and consistently carries out the function with no measurable error.

3.1.2 Endurance

"SkyEye" has a usable flight battery capacity of 10,000mAh. The first 50% of the battery amperage is equal to 70% of the total flight time. The last 50% of usable battery capacity is equal to 30% of the total allowable flight time. This allows approximately 34 minutes of endurance.

Test have shown that the target endurance could not be meet with the available (used) batteries in a safely manner. During one of the flight tests, after a long wait time with system idle on the ground, the batteries reached the failsafe threshold (13V) after only 24min. in the air, which caused the aircraft to land prematurely, before the mission end. This will be corrected by changing the propulsion batteries with fresh new ones, if financial restrictions will allow it, or by reducing idle time; propulsion batteries will have to be connected only a short time before takeoff.

3.2 Payload system performance

3.2.1 Payload Testing

The payload was tested during initial RC flight testing. The weight of the payload was kept at minimum to achieve the desired endurance. Utilizing a qualified RC pilot weight was added to the aircraft until the lower bound of flight time (34min) was reached. The copter was capable of lifting 2lbs, with a MTOW of 6lbs.

Next, gimbal stabilization and vibration dumping was tested, by streaming the images over Wi-Fi while flying at different speeds and orientations. Max. pitch and roll angles for steady shot were found to be 32^{o}, allowing horizontal speeds up to 10m/s (19knots).

3.2.2 Imagery System Testing

The imagery testing began during the final phase of our flight testing. Minor adjustments continue to be made after each test flight to improve image quality. Vibration of the camera is an issue that is constantly addressed and changes to the gimbal have reduced the problem to manageable levels. Operation of the Imagery payload is practiced during every test flight to insure the highest experience level in time for competition.

3.2.3 Target Location Accuracy

According the rule book's key performance parameters the objective is to find the target within 50 feet. To meet this objective the team took target data from practice mission and compared this data to the actual location of the target. The first tests of our target location showed that the location of the majority of the targets were 50ft-100ft from the actual position of the target. After inspection, the team found the camera mount had been 4^{o} lower than the airframes level position. In order to fix this the gimbal calibration setting needed to be adjusted in GCS.

3.3 Autopilot system performance

Flight tests have shown a very good guidance performance in autopilot mode with 2m (less than 7ft) radius on waypoints as can be seen from analyzing telemetry logs of flown missions (Fig. 3.1.) .

Loiter mode proved very satisfactory in winds bellow 5m/s(1kts), and acceptable in stronger winds up to 15m/s (30kts), with a maximum 5m(15ft) position deviation from the programmed waypoint.

Altitude hold is proved to be accurate, within a 1m(3ft) margin.

All Failsafe modes (RC Failsafe, GCS Failsafe, battery Failsafe) proved to work as expected. During test flights done RC Failsafe and GCS Failsafe did not happened in normal operation, but only in forced test mode (RC transmitter shut down intentionally or telemetry link cut on purpose).

3.4 Evaluation results supporting evidence of likely mission accomplishment

After evaluation of the test results presented above and carefully analyzes of many flight logs, team is confident that SkyEye is prepared for accomplishing selected missions. Proof of primary missions' accomplishments can be seen in flight video provided.

Fig. 3.1. Mission telemetry log

4 Safety considerations/approach

In the UPB-FIA UAS department, safety is applied intrinsically. By the use of checklists and redundancy in its systems safety is constantly monitored and addressed. This type of safety carried over to team operations, and is applied to all components of our missions.

For this competition, safety was addressed in three parts: Aircraft, mission and redundancy.

4.1 Specific safety criteria for both operations and design

4.1.1 Aircraft

During the assembly phase of the SkyEye copter, close attention was paid to insure the aircraft was properly assembled as per design. The torque specifications were checked and Loctite was used where applicable. Also on the aircraft, special attention was paid to making sure all loose cables and components are properly secured to reduce the risk of unintended movement during flight operations.

Additional safety precautions that were taken on the aircraft included high visibility paint. Batteries are wrapped in a bright blue to promote high visibility to both operators and onlookers. Before integrating the autopilot system, several external pilot test flights were conducted on the aircraft to insure functionality of the copter, along with payload and endurance tests mentioned in the design portion.

Safety of the operators and the surrounding environment is a prime concern in an autonomous mission. In the event of any equipment failure, the autopilot failsafe is realized in three ways. The first is the RC failsafe that continuously keeps a track of RC radio link, losing which it engages failsafe RTL. The second is a PPM multiplexer which allows the RC pilot to manually over-ride the aircraft any time during flight. The third is the telemetry link that enables GCS operator to take control, transforming the aircraft in a RPV.

Safety is a major concern during any UAS operation. To ensure hassle-free mission execution, every flight crew member has a designated check-list which includes reaction under emergency situations.

During flight operations under normal conditions, the RC pilot receives commands only via the Flight Director.

While the aircraft is in air, the Safety Officer is stationed near the pilot. In case of emergency landing announcement from the pilot, he executes operational procedures to ensure the landing spot is clear and the overall safety of spectators and flight crew.

4.1.2 Mission

A risk assessment tool designed by our safety officer is used before every flight. After completion, this tool gives the operation a go/no go decision based on mission type, environmental factors, and crew readiness. The risk assessment tool relies on a point system where low numbers represent small risk, and larger numbers represent high risks. When added together these points help decide a go/no go decision.

Likewise every mission performed by the SkyEye copter always begins and ends with a checklist. A point is made to never commit anything to memory so that no item is ever missed. Before the flight portion of the mission begins, a crew briefing is held, here the crew assignments are verbalized along with type of takeoff, mission objectives, emergency procedures, and recovery methods. In the event of an emergency or unintended movement, the crew is instructed to take shelter inside, or behind the GCS to prevent being struck by the aircraft.

Crew assignments are a very important part of mission safety. It is imperative that every person participating in a flight test is given an assignment to keep crew focused on the task at hand, and know what their role is in case of an emergency.

The ground control station consists of the parts of the crew who conduct the autopilot and payload operations. It is their job to conduct the checklist before the flight, and commanding operations during flight. The ground station crew is responsible for all operations of the aircraft, unless operations are transferred the external pilot. In the event of an emergency, the GCS crew is required to stay inside the station to prevent injury.

Ground crew members are responsible for the handling of the aircraft when not flying. They are tasked to transport the aircraft, charge batteries and test systems before each flight. Once all checklists are complete, the ground crew must move to inside or behind the ground control station during flight operations.

The safety pilot's responsibility is to handle the external pilot transmitter at all times. When necessary the safety pilot assumes control of the aircraft, and is ultimately responsible for the safety of flight operations. While in flight, the safety pilot operates as a spotter to insure clearance from danger.

The team has established a go/no-go criteria based on previous experiences which are strictly adhered to. SkyEye shall not fly under the following circumstances:

- If there is any precipitation.
- If there is an approaching thunder-storm.
- If the visibility is less than 1 mile.
- If GPS lock fails.

- If there is any perceptible damage during ground operations.
- If range test fails before 200 feet on the primary RC link with all wireless devices operational.
- If winds exceed 20 knots.
- If there is low light.

4.2 Safety risks and mitigation methods

Based on past failures and possible risks experienced from several flight tests, a detailed Failure Modes and Effects Analysis (FMEA) approach was employed to devise a meticulous Risk Mitigation Protocol to be followed by flight crew during flight tests. This practice ensures the system's flight readiness to meet the competitive safety requirements of AUVSI SUAS 2015.

Code Green: Mission continues autonomous
Code Yellow: Mission Continues; Manual override
Code Red: Mission failure; Emergency Landing

Failure	Alarm	Action	If recovered	If not recovered
Imagery (Wi-Fi) link loss	Warning on Imagery terminal	Switch to Loiter	Return to mission	RTL and Land
Telemetry link loss	Warning on GCS terminal	Switch to RTL	Return to mission	RTL and Land
RC link loss	Warning on GCS terminal	Switch to RTL	Return to mission	RTL and Land
Low battery	Warning on GCS terminal	Emergency Land	-	-
Aircraft enters no-fly zone	Warning on GCS terminal	Engage Geo-Fence	Return to mission	Emergency Land
Loss in altitude	Warning on GCS terminal	Switch to Altitude Hold	Return to mission	Emergency Land
GCS terminal crash	Locked or no screen output	Switch to Manual while GCS restart	Return to mission	RTL and Land Manual
Imagery terminal crash	Locked or no screen output	Continue, data stored on-board	Return to mission	Try to end mission
Catastrophic failure	Collision or debris falling	Emergency Land	-	-

4.2.1 Redundancy

Redundancy begins in our ground control station (GCS). The UPB-FIA GCS boasts a triple redundant power system. Since UAS operations cannot be carried out without the GCS, these redundancies are very important. The entire electrical system is powered by external supply power provided. Once that fails the system relies solely on the laptop batteries that also last approximately 90 minutes. So the GCS is capable of operating without the generator for approximately 90 minutes, which is more than enough time considering the flight time of our aircraft is only 34 minutes.

The APM autopilot system also boasts redundancy. During any phase of flight the abort function can be initiated. Depending on what phase the aircraft is in, this feature can safeguard the aircraft and its surroundings. If abort is selected during rotor spin-up or liftoff the rotors will spin down and the aircraft will kill the motor and put itself into prelaunch mode.

At any time during flight operations abort is selected, the aircraft will decelerate to zero airspeed, and begin a controlled decent and land directly below its flight path. Also if battery failsafe condition occurs the same landing strategy will be engaged. This is a great benefit of having an aircraft capable of vertical takeoff and landing (VTOL). Where a fixed-wing aircraft needs a large area for emergency procedures, a VTOL aircraft can simply land in almost any area with little risk to damaging the aircraft or onlookers.

If at any moment communication is lost, the RTL plan programmed into the aircraft for lost communication goes into effect. Before the mission, the operator chooses what the lost communication procedure will be. In our operations, lost communications always results in the aircraft initiating its RTL flight plan, and executing an autonomous landing when arrived at home location. This eliminates any possibility of loss of aircraft due to communication error.