10.2514/6.2016-0743

AIAA SciTech Forum
4-8 January 2016, San Diego, California, USA
AIAA Infotech @ Aerospace

A Dual-Use Unmanned Aerial System for Precision Agriculture and Search and Rescue Applications

2014 NASA Langley Research Center Aeronautics Academy:

Adam Weber[1]
University of Michigan, Ann Arbor, MI 48109

Nathan Lauer[2]
University of Maryland, College Park, MD 20742

Jesse Brady[3]
University of Alaska Anchorage, Anchorage, AK 99508

Callie Branyan[4]
University of Arizona, Tucson, AZ 85721

Mack Goodstein[5]
University at Buffalo, Buffalo, NY 14260

Ateev Gupta[6]
Purdue University, West Lafayette, IN 47907

Brian Katona[7]
Kent State University, Kent, OH 44240

Jonathan Katzman[8]
University of Miami, Coral Gables, FL 33124

Joel Krause[9]
University of Minnesota, Minneapolis, MN 55455

Matthew Siburt[10]
West Virginia University, Morgantown, WV 26506

Melissa Smith[11]
Rensselaer Polytechnic Institute, Troy, NY 12180

Jeff Szczubelek[12]
Wilmington University, New Castle, DE 19720

[1] Undergraduate Student, Aerospace Engineering, akweber@umich.edu
[2] Undergraduate Student, Aerospace Engineering and Computer Science
[3] Undergraduate Student, Mechanical Engineering
[4] Undergraduate Student, Mechanical Engineering
[5] Undergraduate Student, Electrical and Computer Engineering
[6] Undergraduate Student, Mechanical Engineering
[7] Undergraduate Student, Aeronautics
[8] Undergraduate Student, Aerospace Engineering
[9] Undergraduate Student, Aerospace Engineering
[10] Undergraduate Student, Aerospace and Mechanical Engineering
[11] Undergraduate Student, Aerospace Engineering
[12] Graduate Student, Technology and Education

This paper presents the design, construction, testing, and results of the project assigned to the 2014 NASA Langley Research Center Aeronautics Academy. The Academy was tasked with delivering one Unmanned Aerial System (UAS) capable of performing both Search and Rescue (SAR) and Precision Agriculture (PA) missions. The aircraft was constructed using primarily Commercial off-the-Shelf (COTS) electronics and flight hardware housed in a custom-fabricated airframe. The UAS, named TIGRESS (Technology in Ground Rescue and Environmental Stress Sensing) is capable of long-endurance autonomous flight, live video streaming, autonomous detections of persons, creation of Normalized Density Vegetation Index (NDVI) maps, and has completed four successful flights. The results of the flight tests, the design, and the construction methods are presented. Additionally, recommendations for further tests are presented.

I. Nomenclature

α: angle of attack
Λ: sweep angle
ρ: density
a: lift slope coefficient
AR: aspect ratio
b: wing span
c: chord length
C_D: coefficient of drag
C_L: coefficient of lift
d: diameter of propeller
D: drag
***e*:** Oswald's efficiency coefficient
g: acceleration of gravity
L: lift
m: mass
q_∞: dynamic pressure
S: planform area
V_∞: air speed
W: weight

II. Introduction

As technology continues to ceaselessly advance, the utility of the UAS as a platform continues to increase. Unmanned aircraft are becoming more efficient, lighter, faster, less expensive, more accessible, and perform more precisely. All of these factors are opening up UAS for applications outside the military realm not previously imagined. Proposed and realized uses for unmanned aircraft include, but are not limited to, package delivery, infrastructure inspection, wildfire identification, search and rescue, and precision agriculture. The latter two are the subject of this report.

There are countless search and rescue organizations throughout the United States. They exist on the local, state, and federal levels and incorporate both volunteers and professionals alike. One of the most notable search and rescues is the United States Coast Guard (USCG). In 2013 alone, the USCG dispatched 19,420 SAR sorties, which lead to the successful saving of 3,753 lives. However, 226 lives were lost after the Coast Guard had been notified[1]. Because human personnel carry about USCG rescue missions, safety conditions place limitations on the operations. Operating procedures dictate that deployment of aircraft should be avoided in the presence of low cloud ceilings, thunderstorms, inclement weather, and high turbulence[2]. Not coincidentally, these conditions also create the situations that necessitate a SAR mission in the first place, yet they limit rescue personnel's ability to operate. In such situations, a UAS is clearly practical. Because it is inherently unmanned, they can be deployed without the further risk of human life. UAS can be used in place of, or in supplement to, manned SAR missions.

Damage to agricultural crops in the United States leads to tremendous economic consequences. Between the years 2001 and 2010, there was an annual average of $4.1 billion loss of crops, spiking to $17.3 billion in 2012[3].

These losses are due to drought, heat, insect infestation, poor nutrients, or some combination of these factors. This is not only economically detrimental to farmers, but has impacts on populations, food consumers, and livestock industries. UAS equipped with remote sensing capabilities are able to scan entire farms and fields to pinpoint the exact location of damaged or stressed stands of crops that may not be visible to the naked eye—known as Precision Agriculture. This allows for precise application of pesticides, fertilizer, or water. In addition to mitigation of potential crop losses, there are secondary benefits. Fertilizer and insecticide use, and thus runoff into water supplies and damage to ecosystems, is limited. Also, water use can be minimized, which may prove especially beneficial given the current state of the drought throughout the American West.

In the current market, there are commercial UAS solutions to each of these two mission profiles. However, they are prohibitively expensive and inaccessible to the small farmers and SAR outfits that could most benefit from these technologies. Additionally, there is an overlap between the hardware, software, and sensors necessary for both SAR and PA. The work presented in this paper seeks to develop a low-cost UAS that can perform both SAR and PA mission profiles. The completed product is shown in flight below in Figure 1.

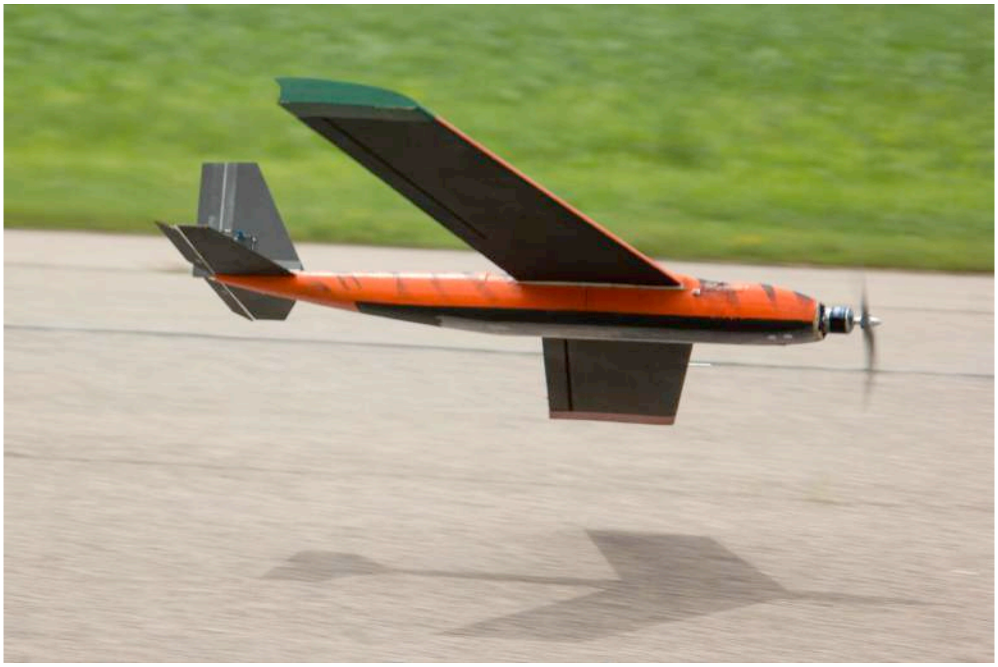


Figure 1: TIGRESS in Flight

III. Design Overview

The design of TIGRESS was driven by a set of requirements levied at the initial project conception. Each mission has threshold and objective levels of success intended to capture the flight profiles the UAS must perform. For Search and Rescue, the threshold level of success was to travel two kilometers from the point of takeoff, search a four square kilometer area, stream live video, and downlink infrared images of the search area; the objective level of success was to autonomously detect and locate persons in this search area. For Precision Agriculture, the threshold level of success was to take still images of a one thousand acre field, discern stressed plants to a resolution of a one square foot area, produce an NDVI map of the field, and autonomously locate stressed plants. An additional overarching requirement of the project was to make the UAS a low-cost product.

All of the previously described requirements were considered when performing the trade studies to design TIGRESS. The final airframe design is a hand-launched, belly-landing, high wing, single motor, conventional fixed-wing, and electrically powered aircraft. The CAD mockups are presented below in Figure 2 and Figure 3.

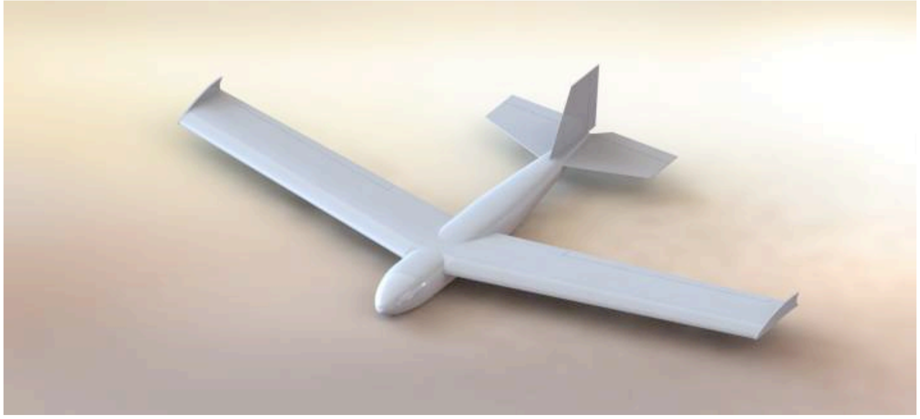


Figure 2: TIGRESS Fully Assembled

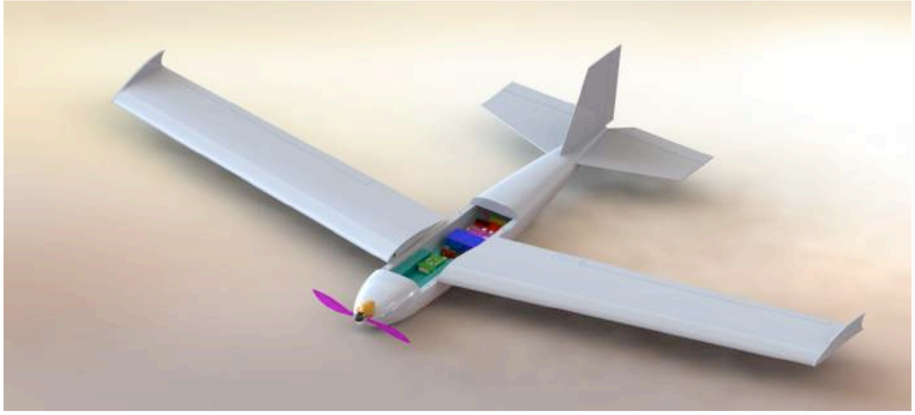


Figure 3: Tigress with Hatch Removed

The UAS was designed to be hand-launched and land on its belly, as it is unlikely that a long runway or road will be accessible in a SAR or PA situation, and it reduces takeoff weight. Consequently, the wings are high mounted to generate ground clearance and prevent damage during landing. A conventional tube-and-wing design was selected over a rotorcraft configuration to comply with the long range dictated by the requirements, and to enable higher cruise speeds in SAR sorties when time is of the essence. Lastly, the powertrain was designed to be electric because batteries are more efficient than liquid propellants, will enable long duration flight, are standard for use on small UAS, and do not generate the excess noise, heat, and vibration associated with gasoline motors.

The flight computer, payload electronics, sensors, actuators, and all other flight components were selected to be commercial off-the-shelf (COTS). Following a preliminary literature review, it was decided that the missions could be accomplished using commercially available solutions. Because the project had to be completed in ten weeks, this was extremely beneficial to the team as it eliminated the need to develop any components from scratch. Additionally, development is expensive; utilizing COTS hardware meets the low-cost objective for the project. The airframe was custom designed and constructed specifically to meet the mass and volume requirements of these components selected to satisfy the mission objectives.

IV. Airframe Design

The design of the airframe was performed analytically using initial estimates of metrics such as total weight of the completed UAS.

The density of foam used to construct the body is roughly 30 oz/ft^3, making the airframe weight 12 oz. From previous fabrication experience, the fiberglass was estimated to add another 6 oz. Thus, the estimated airframe weight was 1.2 pounds. Including weight estimates from the electronics and propulsion systems, the final weight estimate was 10 pounds.

The first design iteration was modeled after high endurance planes, such as the U2. The UAS was designed with a 6-inch chord and 7-foot span. This design was mainly aesthetic. Calculations were then made to ensure that this design was feasible. Below is a calculation for the required C_l for the plane to fly at 25 mph.

$$
\begin{gathered}
W = 10 lbs \\
V_\infty = 25 mph = 36.7 \frac{ft}{s} \\
S = 3.5 ft^2 \\
\rho = 0.075966 \frac{lbs}{ft^3} \\
q_\infty = \frac{1}{2}\rho V_\infty^2 = \frac{1}{2} * 0.075966 * 36.7^2 = 51.1591 \frac{lbs * ft}{s^2} \\
L = mg = 10 * 32.2 = 322 \frac{lbs * ft}{s^2} \\
C_L = \frac{L}{q_\infty S} = \frac{322}{51.1591 * 3.5} = 1.798 \qquad \text{Equation 1}
\end{gathered}
$$

Therefore a C_L of 1.8 was required to fly at the desired flight conditions. However, most NACA airfoils stall below that point; the designed wing would not support the aircraft. To fix this issue, the surface area of the wing, and the flight speed were increased. The wing was changed to a tapered wing, with a 12-inch chord at the root to a 10-inch chord at the tip.

$$
\begin{gathered}
W = 10 lbs \\
V_\infty = 30 mph = 44 \frac{ft}{s} \\
S = 6.466 ft^2 \\
\rho = 0.075966 \frac{lbs}{ft^3} \\
q_\infty = \frac{1}{2}\rho V_\infty^2 = \frac{1}{2} * 0.075966 * 44^2 = 73.5 \frac{lbs * ft}{s^2} \\
L = mg = 10 * 32.2 = 322 \frac{lbs * ft}{s^2} \\
C_L = \frac{L}{q_\infty S} = \frac{322}{73.5 * 6.466} = 0.68 \qquad \text{Equation 2}
\end{gathered}
$$

This C_L was more realistically obtainable. In order for the UAS to fly as close to level as possible, an airfoil that reaches approximately $C_L = 0.7$ at a near 0^o angle of attack with a high stall angle was ideal. Searching through roughly 100 different airfoils yielded the S4180 airfoil as the best choice. It has a low Reynolds number, reaches $C_L = 0.7$ at an angle of attack of 2.5^o, and stalls at 14^o.

In terms of longitudinal placement of the wings on the fuselage, the aerodynamic center should be aft of the center of gravity for greater stability. To find the center of gravity, a more detailed CAD model of the UAV was made. The model showed that with the current design, the center of gravity was behind the aerodynamic center. To fix this issue, a 15^o sweep was added to the wings. This not only moved the ac back, but also helped to decrease aerodynamic drag.

In terms of appropriately sizing the horizontal and vertical tails, a general rule of thumb derived from commercially available models is:

$$
\frac{V}{h} = \frac{S_h l_c}{Sc} \qquad \text{Equation 3}
$$

This formula defines the appropriate ratio of the surface areas of the wing, the horizontal tail, the chord length, and the distance between the aerodynamic center and the quarter chord of the horizontal tail for small UAS. This ratio should be between 0.3 and 0.6. The tail was designed to have a root chord of 9 inches, tip chord of 4 3/8 inches, and span of 2 feet. Below is the calculation for the designed tail.

$$S = 6.466\,ft^2$$
$$S_h = 1.1178\,ft^2$$
$$l_c = 27.98in$$
$$c = 11in$$
$$\frac{V}{h} = \frac{1.1178 * 27.98}{6.466 * 11} = 0.44$$

This value is within the accepted region of V/h values. The calculation for the vertical tail is very similar, though accepted values are between 0.02 and 0.05.

$$\frac{V}{v} = \frac{S_v l_v}{Sb} \qquad \text{Equation 4}$$

The vertical tail had the same tip and root chord as the horizontal tail, but the height was only 7 inches.

$$S = 6.466\,ft^2$$
$$S_v = 0.6503\,ft^2$$
$$l_v = 27.98in$$
$$b = 84in$$
$$\frac{V}{v} = \frac{0.6503 * 27.98}{6.466 * 84} = 0.034$$

This is also within the region of accepted values. Similar to the horizontal and vertical tail design, there are only general rules of thumb for control surfaces sizing. The aileron area should be roughly 12% of the wing area.

$$S_{ail} = 0.12 * 462in^2 = 55.44in^2 \qquad \text{Equation 5}$$

Therefore, the ailerons are 2 x 27 inches.

The rudder should be 30% of the tail area.

$$S_{rudder} = 0.30 * 46.8in^2 = 14.04in^2 \qquad \text{Equation 6}$$

The rudder is 2 x 7 inches. This is the full height of the tail.

The elevators should be 25% of the horizontal tail area.

$$S_{Htail} = 0.25 * 80.6in^2 = 20.15in^2 \qquad \text{Equation 7}$$

The elevators are each 2 x 11 inches. This is an inch short of the full span on each tail. This allows room for the rudder to move back and forth unimpeded.

V. Power System Design

Selecting the necessary COTS components for the power system was done by a combination of numerical analysis and following standards from the radio-controlled aircraft industry. The integral parts are: an electric motor, and electronic speed controller (ESC), servo motors for the control surfaces, and lithium polymer (LiPo) batteries.

Motor selection was based on the following assumptions: the aircraft would be hand launched and it would weigh on the order 10 of pounds. E-Flite, a motor manufacturer recommends that for a ten pound vehicle the minimum power requirement is one thousand watts[4]. After researching many electric motors, the E-Flite Power 52 Brushless Out-Runner Motor was chosen because it has the most appropriate power to weight ratio, and the lowest continuous current draw while generating 1,650 Watts of power. In addition, a brushless out-runner motor has increased longevity over a brushed in-runner motor because there is less friction between multiple moving parts.

E-Flite recommends that an 80 amp ESC and a 14x8 propeller be used with this motor. Based on these choices, Equation 8 was used to calculate the thrust at varying speeds:

$$F_T = 1.225\frac{\pi(0.0254d)^2}{4}[(RPM_{prop}0.0254\,pitch\frac{1\min}{60\sec})^2 \\ -(RPM_{prop}0.0254\,pitch\frac{1\min}{60\sec})V_0](\frac{d}{3.29546\,pitch})^{1.5} \qquad \text{Equation 8}$$

At take-off, where the velocity of the vehicle is zero, the motor generates 7.6 pounds of thrust. At cruise, where the velocity of the vehicle is thirty miles per hour, it generates 3.8 pounds of thrust. At the operating altitude of the aircraft, changes in density are negligible. Therefore, the density of air was approximated to that of sea level: 1.225 kilograms per cubic meter. Using Autodesk Flow Design, the coefficient of drag was calculated to be 0.36. Finally, the cross-sectional area of the body was 0.67 square meters. Therefore, using Equation 9 the body of the aircraft at thirty miles per hour had 0.8 pounds of drag.

$$F_D = \frac{1}{2}\rho v^2 C_D A \qquad \text{Equation 9}$$

In order to ensure that the chosen servos would be able to move the control surfaces under the most extreme conditions, the servos were tested using worst-case estimates for weight, power and performance.

Proper aircraft control requires the control surfaces to comprise approximately twenty to thirty percent of the wing area. Thus, the ailerons were approximated to a length of 1.5 inches and an 18-inch span. The forces on the aileron were calculated in order to determine the necessary torque requirements for the servo. As given by a drag coefficient table in a database the best and worst cast drag coefficients were 0.04 and 1.15, respectively. Equation 10 yielded the corresponding lift coefficients of 1.110 and 5.95, respectively. Equations 11 and 12 yielded the total lift and drag of the aircraft. The estimated arm length of the servo was 0.75 inches. Using equation 13, the best and worst case minimum torque requirements were calculated.

$$C_L = \sqrt{C_D \pi e AR} \qquad \text{Equation 10}$$

$$D = \frac{1}{2}\rho V^2 S C_D \qquad \text{Equation 11}$$

$$L = \frac{1}{2}\rho V^2 S C_L \qquad \text{Equation 12}$$

$$T = D\sin\theta + L\cos\theta \qquad \text{Equation 13}$$

Angle (°)	Best Case Torque (Newton-meters)	Worst Case Torque (Newton-meters)
0	0.04144	0.222
30	0.0367	0.214
45	0.0304	0.1876
90	0.0015	0.0429

Table 1: Torque Scenarios

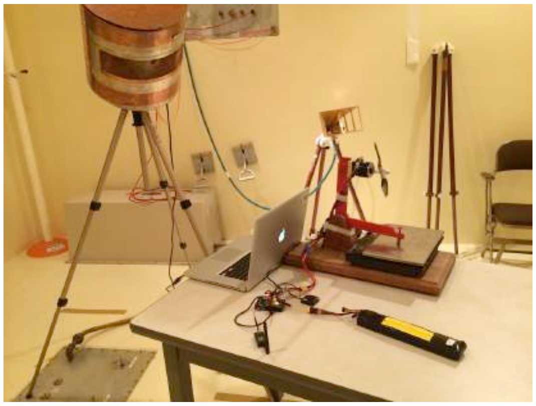
The HS 5125 MG Digital Wing servo was chosen because it meets and surpasses the torque requirements for all scenarios. In addition, it has a slender body shape that allows for easy mounting in a thin wing. The benefits of the torque, shape, and mounting capability outweigh the extra mass of this servo in comparison to other servos considered.

For the Power 52 Brushless Out-Runner Motor, E-Flite recommends a four to six cell lithium polymer ion battery with a 6000 milliamp-hour capacity. In aerobatic flight, a battery with these characteristics provides an estimated twenty to twenty-five minutes of flight time. Therefore, to be able to execute the mission profile, which requires up to ninety minutes of straight and level flight, multiple such batteries are required.

To accommodate the mission profile, the Thunder Power RC Pro Lite MS battery was selected. It is a four-cell battery with an 8000-milliamp hour (mAh) capacity. Wired in parallel, three of these batteries easily provide ninety minutes of flight time.

Finally, based off the power calculations, the Thunder Power RC Pro Lite MS 4,000 mAh 3S battery was chosen to separately power the electronics.

Several different propellers were tested with the selected motor, ESC, and batteries. The motor and prop were mounted on a thrust stand resting on a digital scale, as shown in Figure 4 below. The propulsion system was taken to full throttle for each of the different subject propellers, and the maximum thrust was recorded.



Diameter (inches)	**Pitch (inches)**	**Thrust (lbs)**
10	6	2.8125
12	6	5.8125
14	8	7.875
15	6	9.4375

Figure 4: Propulsion System Testing (Left) and Results (Right)

VI. Electronics Payload

The electronics payload of TIGRESS consists of the sensors (primarily cameras), flight controller/autopilot, multiplexer, onboard computer, and video streaming hardware. The electronics are powered separately from the propulsion system to remove single points of failure. All of the hardware was selected specifically to meet the requirements of the mission.

Creating an NDVI map requires a camera with the capability of sensing near infrared light in addition to the normal RGB channels sensed by a traditional consumer camera. Cameras of this type are often referred to as multispectral cameras, due to their ability to sense wavelengths outside of the visible light spectrum. Many custom made multispectral cameras specifically designed for vegetation indexing exist on the commercial market. These cameras offer specially built filters, and are optimized sensors for producing false-color composite images such as NDVI maps. However, it has been shown[5] that consumer point-and-shoot cameras modified from their original format can be used as multispectral cameras. By removing the near-infrared blocking filter typically installed by manufacturers, and installing another filter passing the appropriate wavelengths, it is possible to allow one of the channels of the camera's sensor (typically the red channel) to absorb near-infrared light. This expands the consumer camera's capabilities to mimic those of purpose-built multi-spectral cameras.

The Canon SX260 HS digital camera, a point-and-shoot camera with a 12.1 Megapixel CMOS sensor and a 25mm wide-angle lens, is the camera that best fits the NDVI mapping needs. The infrared conversion is commonly done on this camera, both by hobbyists and by organizations that sell modified multi-spectral cameras. This camera is capable of recording photos with location metadata, and has a built-in GPS receiver. For the purpose of aerial imaging, the SX260 supports Canon Hack Development Kit, a software application that allows users to modify most of the settings on a Canon camera, and facilitates applications such as remote triggering. The camera is capable of capturing 2.4 images per second, which is sufficiently fast for capturing multiple images in the air. The camera is also natively supported by Mission Planner software that controls the selected autopilot.

The central component of the UAS is the autopilot, which performs all of the navigation and control functions of the UAS. There are many UAS autopilot systems offered on the market, some developed by corporations and others developed under open source or open hardware agreements. The 3DR PixHawk was chosen for its high level of development and the ease of interfacing it with other components. 3D Robotics sells an off the shelf version of an open-hardware design from ETH Zurich called the PixHawk computer vision project. It can control up to 14 servo channels as well as peripheral equipment through multiple interface types. It is offered with a digital airspeed sensor, telemetry radio, and GPS/Compass unit. The PixHawk is also supported by Mission Planner software, which allows for waypoint planning, surveying, and photo triggering. In addition, the PixHawk is offered at a price that is affordable for the target users of the UAS.

A servo multiplexer allows a switch from autopilot control to RF transmitter control in order to ensure the safety of the aircraft and its environment in case of an autopilot failure. The 8-channel Acroname RxMux is explicitly designed for use with autonomous vehicles. The multiplexer operates in two states. In state A, the multiplexer passes input from the autopilot to the servos. In state B, the multiplexer passes input from the RF receiver to the servos. In essence, the multiplexer allows the user to take manual control of the UAS at any time to mitigate any potential system failures.

The mission profile requires an independent on-board processing system. To properly use the NDVI software, the precision agriculture aspect of the mission needs to store still images. The search and rescue aspect requires actual processing power to analyze video and image feeds in order to identify humans. A number of computers were considered, including the BeagleBoard, Arduino Uno, and the Raspberry Pi.

The Raspberry Pi is a light (40g) and compact computer, with 512 MB of RAM and a 700 MHz processor. Using the MAVlink communications protocol, it can interface with the PixHawk autopilot and the Mission Planner software. The Raspberry Pi is also capable of receiving GPS signals through an external GPS receiver, providing a backup to the GPS available on the PixHawk autopilot. A specially designed small digital camera comes with the Raspberry Pi, in both normal color and near-infrared configurations, making it ideal for agriculture and search and rescue applications. Finally, the video RCA output on the Raspberry Pi makes it possible to broadcast video taken from one of the on-board cameras to a ground station. The small size, lightweight, compatibility, and ease of use make the Raspberry Pi ideal for the mission profile, and therefore was chosen as the on-board computer.

In order to transmit a live video feed from the aircraft to the ground, a 1000mW, 900 MHz transmitter was chosen. This video transmitter is capable of transmitting video captured by any camera connected to the Raspberry Pi to a ground-based receiver. The video transmitter interfaces with the Raspberry Pi through the Raspberry Pi’s RCA output.

A CAD mockup of the electronics payload is provided below in Figure 5. This also includes the ESC and LiPo batteries for the power system (the large rectangular prisms towards the bottom of the payload.

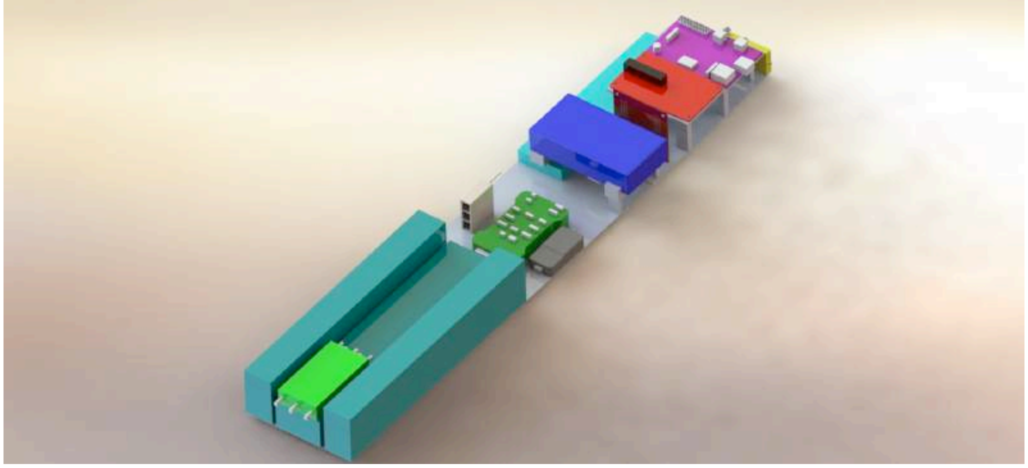


Figure 5: All Payload Hardware of TIGRESS

VII. Discussion of Technology

NDVI

The Normalized Difference Vegetation Index is a method of evaluating the health of vegetation based on the reflectivity of the plant material being assessed. The health of plants is based on the absorption of sunlight by chlorophyll. Chlorophyll reflects wavelengths in the near-infrared domain very highly, while it reflects very little red wavelengths. Therefore, healthy plants will reflect large amounts of infrared wavelengths relative to the amount of red light they reflect. Cameras capable of sensing wavelengths in both the infrared and visible light spectrums can document vegetation health. By post-processing an image captured by one of these cameras, it is possible to create an indexed image displaying the relative health of plant material. Combining multiple images leads to the creation of a map showing the relative health of multiple plants, a field or an entire region. Traditionally, an NDVI image[6] is created by applying the following formula to each pixel of an image:

$$NDVI = \frac{NIR - R}{NIR + R} \qquad \text{Equation 14}$$

Where NIR is the intensity of near IR light captured by the sensor and R is the intensity of red light captured by the sensor. The result is a value between -1 and 1, with 1 representing the greenest vegetation. A grayscale image composed of pixels corresponding to these values creates a visualization of the health of the plants in the image. Applying false color to the image through look-up-tables results in a readable mosaic of vegetation health data.

The traditional formula for NDVI was designed for use on satellites, and collected wavelengths in the red and infrared spectrums to avoid scattering from Rayleigh light scattering in the blue channel. Flying a sensor suite on a UAV at low altitudes makes the Rayleigh light scattering effect[7], and therefore the blue channel can be used in the creation of the NDVI. Using all three bands of light collected from a multi-spectral camera may allow for an improved index. Including light from the green channel, and compensating by doubling the weight assigned to the blue channel results in an enhanced NDVI, represented as:

$$ENDVI = \frac{NIR + G - 2B}{NIR + G + 2B} \qquad \text{Equation 15}$$

The algorithm for computing NDVI and ENDVI was developed using Fiji ImageJ image processing software. Three methods were evaluated: the Photo Monitoring tool available as a plug-in for ImageJ, a custom-made script for traditional NDVI, and a custom-made script for Enhanced NDVI. Each script begins by taking an RGB image from the modified camera, in which the red channel has absorbed infrared light, the green channel has absorbed green light, and the blue channel has absorbed visible light. The RGB image is split into three individual channels, and then combined according to the equations above. The resulting image then has a lookup table applied to it, which applies colors to the resulting image according to the NDVI values of the pixels. The example images in Figure 6 illustrate areas where the NDVI map successfully identifies stressed plant material. The stressed leaves on the plant (connected by yellow lines) in the image show up in a deep red color in the NDVI map on the right. The ground also appears red, as it is inorganic.

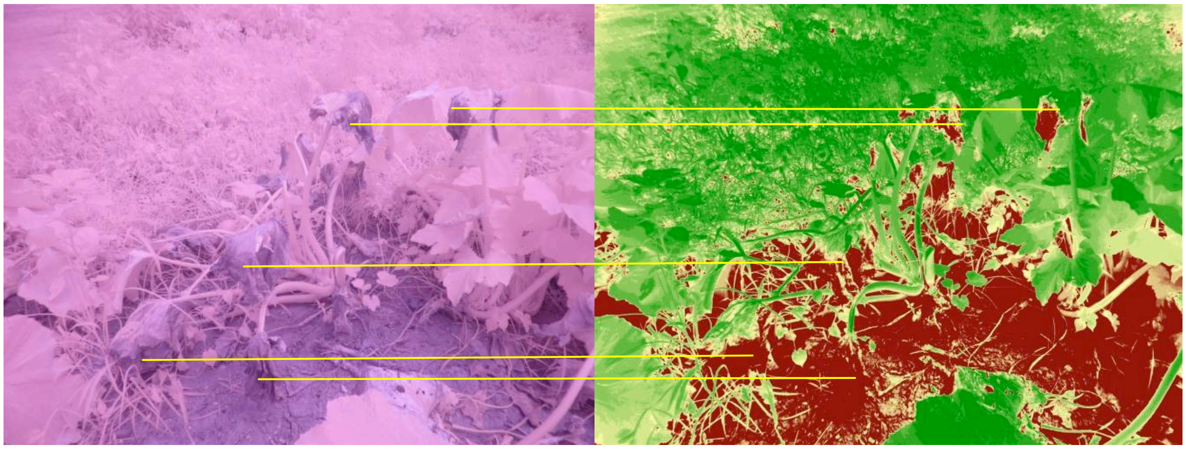


Figure 6: Unprocessed near-infrared and processed ENDVI mapping of plant with stressed leaves.

SAR

TIGRESS supports Search and Rescue operations through two primary methodologies: video downlink of Raspberry Pi (RPi) camera's field of view, and processed images identifying salient objects. The video allows ground operators to see what TIGRESS is seeing, in order for the operators to have a continuous view of the relevant area. The processed images identify salient objects, or "could be humans," to make human detection simpler. The processing happens completely autonomously, and therefore the processed images are easily viewable across multiple computer platforms.

TIGRESS only broadcasts video, with no data uplink, and therefore the ground operator is only able to view the content of the RPi computer's screen. The screen is divided in half: the left half displays video of the terrain below, and the right half displays the previous four processed images, with red boxes around identified salient objects. Each of these images is written to a file on RPi's SD card, where they can be easily accessed after the flight.

On the UAS the RPi and Pixhawk are completely separate systems, and do not communicate with each other. Therefore, the search and rescue software begins on startup, when power is initiated. The software is composed of two independent programs. The first program displays live video feed using Raspistill, the software associated with still images on the RPi camera. Every four seconds an image is written to a file to be processed. The second program processes each of those images, draws red boxes around each identified salient object, displays the processed image, and writes it to a final file. The final file can be accessed via the RPi's SD card, for further viewing and analysis.

The algorithm to identify salient objects is composed of six discreet steps. In the description below, an example image is displayed at multiple steps to show how the algorithm works. RPi has software support for multithreading, allowing the time to process a single image to improve significantly. The Software is written in C++, relies heavily on Intel's OpenCV library, and is based off work conducted by Toby Breckon and Jan Sokalski, Cranfield University[8], and Rebecca Johanning from NASA JSC.

Step 1

Four images are read into the program, and put into a vector of unprocessed images. The number four is not arbitrary; four processed images displayed on the right half the RPi's screen maximize processing speed and screen area per image. GPS data is also read in to define various function parameters based off altitude. Four threads are created, and each one grabs a single image from the unprocessed vector. The remainder of the steps describe how a single image is processed even though the multithreading allows for four images to be processed at once.

Step 2

The input image is put through a process known as mean shift segmentation. Essentially, mean shift segmentation blurs the image. This is advantageous for two reasons. First, it helps to eliminate noise. Second, it removes small, sharp lines to help separate the image into larger sections. For example, blades of grass become a single large area of green, instead of hundreds of independent objects.

Step 3

The blurred image is split into eight separate color channels. Most images are stored in three parts: a section defining how much blue there is in a given pixel, a section for green, and a section for red. A single color channel is an image that stores only the red component, without the blue and green components. The eight color channels are grayscale; blue, green, and red normalized by intensity; and four broadly tuned channels. These broadly tuned channels are combinations of the above, and are defined as such:

$$R = red - green + blue^2 \qquad \text{Equation 16}$$

$$G = green - red + blue^2 \qquad \text{Equation 17}$$

$$B = blue - green + red^2 \qquad \text{Equation 18}$$

$$Y = red + green - |red - green|^2 \qquad \text{Equation 19}$$

The purpose behind this is to emphasize certain parts of the image. In each different color channel, some parts of the image appear significantly brighter than the original image, and other areas are damped. It is therefore extremely likely that the human would appear brightly in one of these channels.

Step 4

Edges are calculated for each of the eight color channels. Edge data is gathered using a gradient method, and the canny edge detector. The gradient method creates a gradient field of pixel data, and marks an edge in areas of large, rapid change. The canny edge detector splits image data into areas of high, medium, and low pixel change. High change is marked as an edge, low change is ignored, and medium change is marked as an edge only if that pixel is connected to an area of high change. For each method, all the eight color channel images with edge data are gathered into a single image, yielding a gradient edge image and a canny edge image. Those two images are then multiplicatively combined into a single image.

Step 5

A hierarchy of contours is generated for the canny-gradient edge image from the previous step. Contours are essentially connected edges, which then allow the software to define shapes. Any contour that is approximately the size of a human at the given altitude is boxed in red on the original input image. This is the final image with detected salient objects boxed.

Step 6

The final image is written to a file and displayed on the screen, such that the ground operators can view it. While being displayed, the algorithm begins again, reading in four more images from the input file. The entire algorithm continues while the RPi camera is taking pictures and TIGRESS is operating at an altitude greater than 25 ft.

VIII. Construction

Because the mission requirements put an emphasis on portability and minimizing weight, construction was performed using materials that are both strong and lightweight. The primary materials are foam, fiberglass, carbon fiber, and wood.

The wings are constructed out of Owens Corning Foamular insulation. This product is strong in compression, tension, and bending and has a very low density. The wings were wrapped in fiberglass to add additional strength and to protect them from contact with the ground during belly landings. They were cut using a computer-controlled hotwire cutter specifically designed for wing fabrication. For portability, the wings are mounted to the removable hatch using carbon fire spars for further increased strength in bending. This can be seen below in Figures 7 and 8.

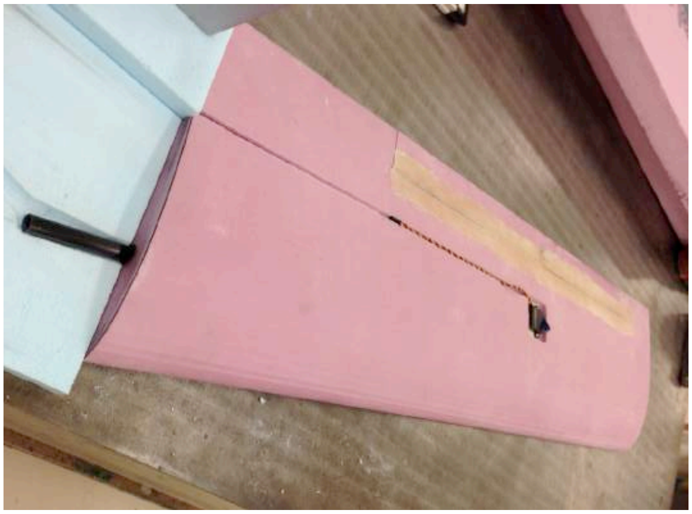


Figure 7: One Completed Wing

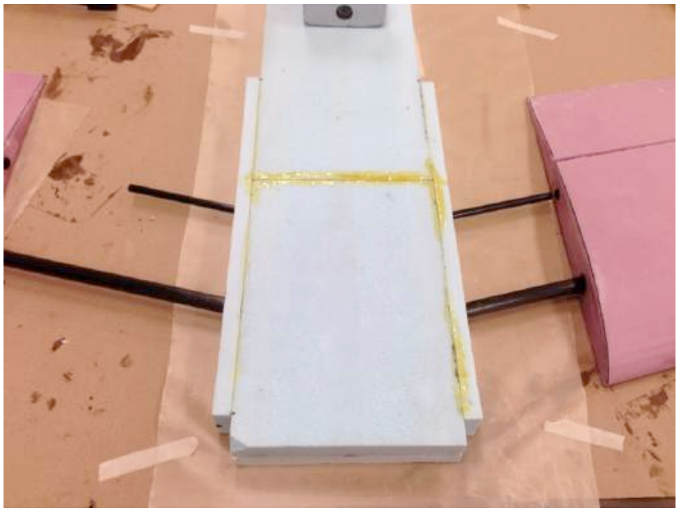


Figure 8: The Carbon Fiber Spars Used to Mount the Wings

Similarly, the fuselage is constructed out of Foamular wrapped in fiberglass. The CAD file was fed to a CNC machine, which carved a block of material to the exact specified dimensions. This included the hollowed-out cavity that houses the electronics and propulsion payload.

The horizontal and vertical stabilizers were constructed as flat plates from basswood and then wrapped in carbon fiber. They were mounted into slots on the tail of the fuselage and held fast using epoxy.

The control surfaces were cut directly from the wings and stabilizers. The ailerons were mounted to the wings using a strip of Kevlar bonded with epoxy, and the elevators and rudder were attached to the horizontal and vertical stabilizers, respectively, using tape in a similar manner. This eliminates the gap typically present aft of a control surface, thereby lowering the overall drag of the UAS. The aileron servo can be seen above in Figure 7, and the empennage servos are shown below in Figure 9.

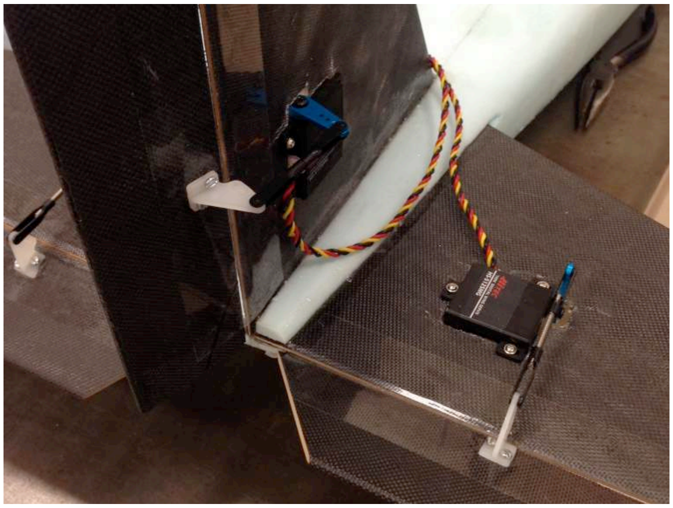


Figure 9: Empennage at Tail of UAS, Control Surfaces, and Servo Motors

The servos used to actuate the control surfaces were mounted directly into cavities on the wings and stabilizers, also to minimize drag. The motor is mounted to a wooden plate on the nose of the fuselage, and secured using bolts that run into the electronics cavity. The hatch and wings are secured to the fuselage using screws, allowing for quick access to the payload electronics.

The airframe is presented, completed, below in Figure 10 and Figure 11. This initial prototype cost on the order of $5000 to manufacture.

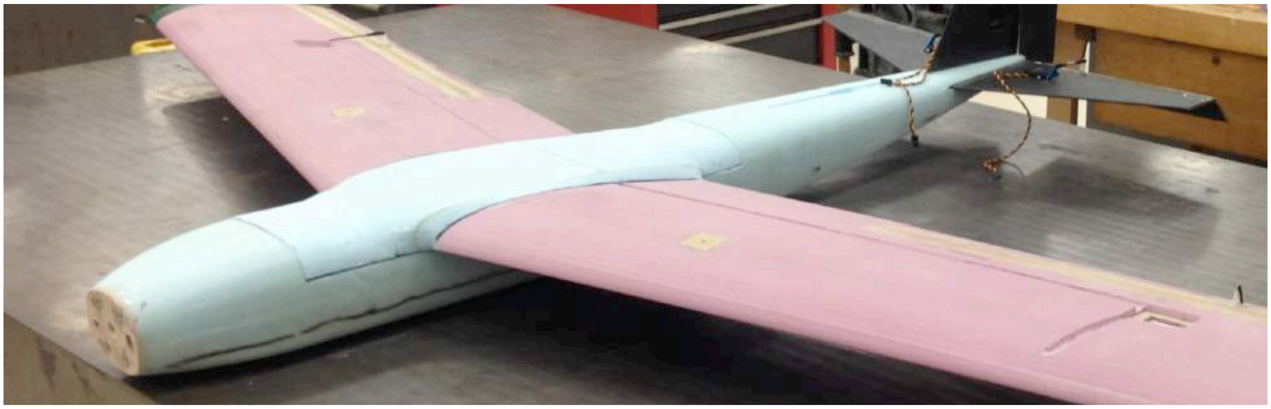


Figure 10: Completed, Unpainted Airframe

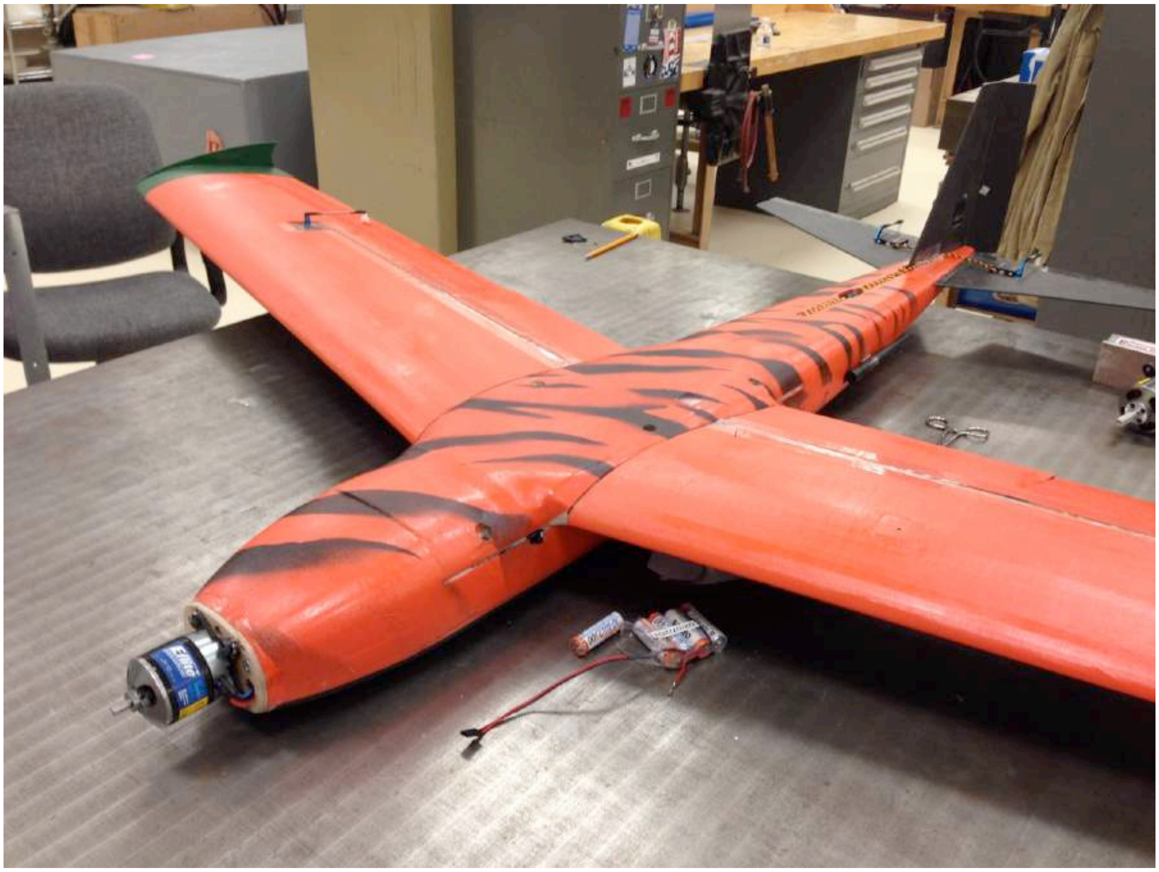


Figure 11: Completed and Fully Integrated Airframe

IX. Results

Search and Rescue

As indicated in the mission statement, there were three goals for search and rescue. First, fly 2 kilometers, search a four square kilometer area, and fly back 2 kilometers to the ground station. Second, the threshold level of success, transmit live video in the infrared range of the search grid. Third, the object level of success, achieve on-board autonomous detection of people in the search area.

Due to limitations on available flight time and restrictions in the Flight Safety Release (FSR) issued by NASA Langley, a full flight demonstrating the flight parameters was not conducted. However, flight four used just 8% battery, while flying at approximately 50 knots for six minutes and sevens seconds, in a wind calm environment. The total distance required to fly to properly search the area, with the transition to and from the search area, is 99 kilometers at an altitude of 400 ft. At the above speed, that would require 64 minutes to search. With the above current draw, in 64 minutes TIGRESS would use 84% of the total battery life. Therefore, there is evidence to indicate that the aircraft is capable of achieving this mission parameter.

The threshold level of success was achieved. Live video in the infrared range was transmitted from TIGRESS to the ground station in all of the flight attempts, except for the fourth attempt. However, the fourth attempt had a power loss at the ground station due to technical difficulties with the trailer, and did not represent a failure on the part of TIGRESS.

True autonomous detection of humans is incredibly difficult, and was not achieved. However, near-autonomous detection was achieved. Instead of identifying humans, the software identified salient objects that are human sized, and boxed them in red. These images were successfully transmitted back to the ground station where the ground operator analyzed the identified salient objects to identify humans and false positives. In Figure 12 below, the unprocessed image is shown on the left and the processed image is shown on the right. The red box towards the center of the image is a member of the team on the ground; the one towards the right is a false positive.

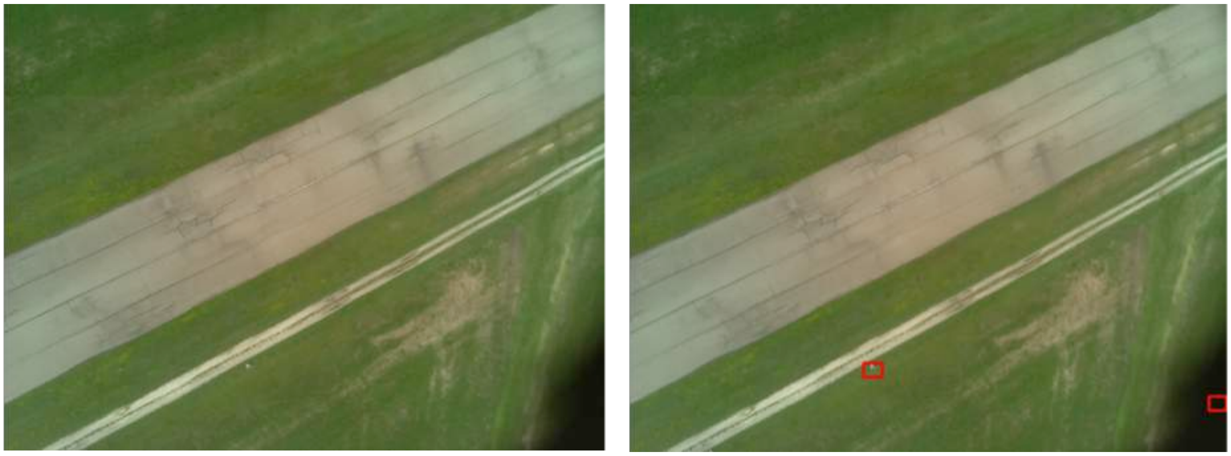


Figure 12: Unprocessed (Left) and Processed (Right) SAR Images Streamed to the Ground Station

Precision Agriculture

During flight-testing, successful near-infrared images of the airport were gathered from the Canon camera. The camera was set to take pictures at a designated interval using CHDK software. Images were captured in the air, and stored on the camera's internal memory card. The images were then post-processed using a custom-made script in ImageJ. The enhanced NDVI formula was used to process the images. Because these were initial test flights and the exact cruise speed and flight path of the aircraft were not known, a composite map of the images could not be made. In addition, the inability of the plane to fly completely autonomously during the test flight prevented the plane from autonomously located stressed areas.

However, the captured images successfully differentiated healthy plants, unhealthy plants and inorganic material. The crops in the fields surrounding the airport appeared green in the NDVI images, while the runway and roads appear dark red. See the images in Figure 13. Discernible areas of unhealthy plant material, such as tire tracks in the fields, appear as yellow areas in the images. Many of the images were affected by shadow, especially in areas where there were trees. Areas with shadows appear redder than nearby areas in more direct sunlight. The images also appear to be affected by some distortion, most likely from the glass protecting the lens, and the tape used to protect the bottom of the aircraft during landing. This appears in the image as red vignettes surrounding many of the images. These vignettes did not appear in every image, making the source of the distortion difficult to diagnose.

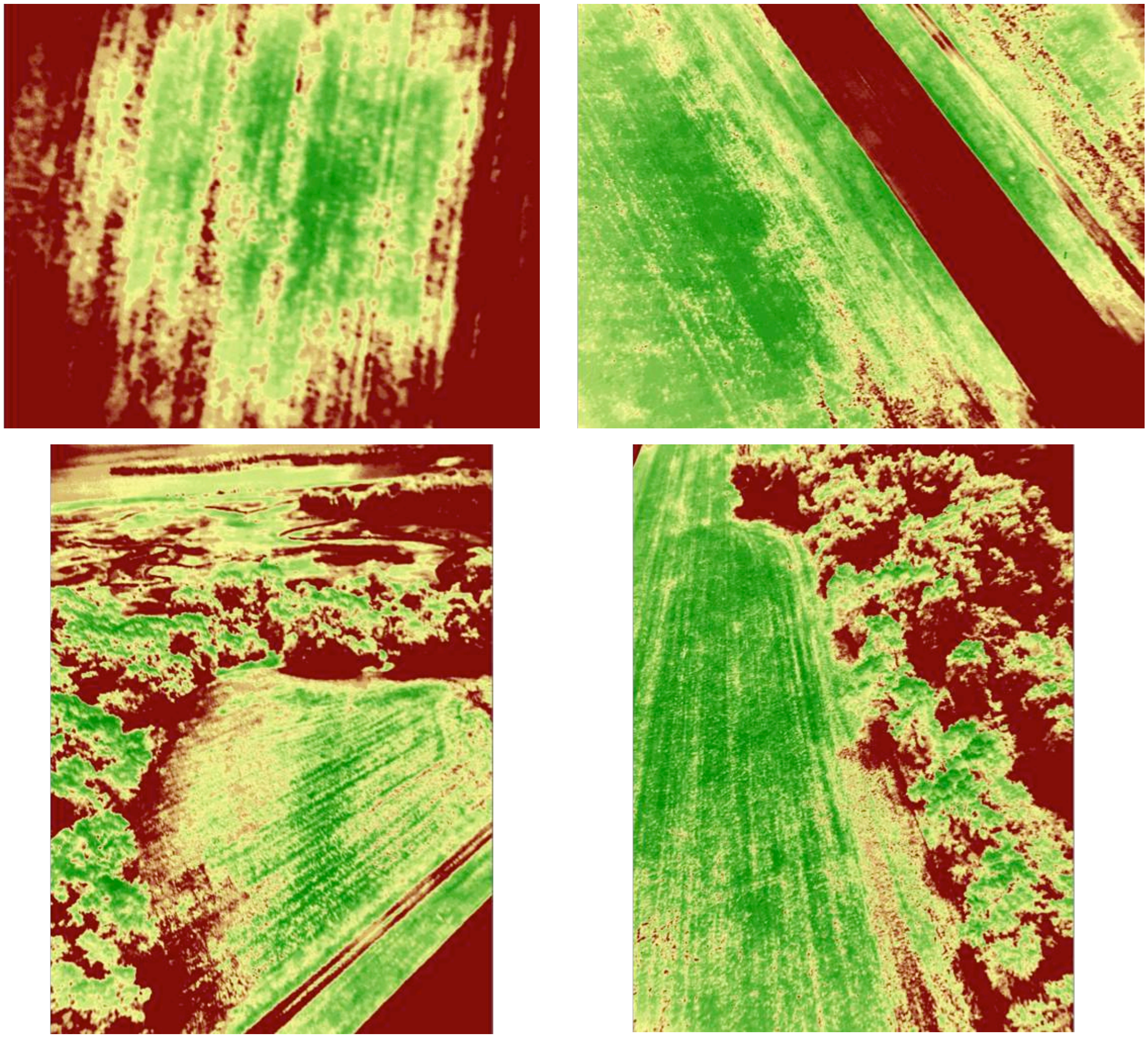


Figure 13: NDVI Image Results from Flight Test

X. Future Work

Now that the completed UAS has undergone initial airworthiness flights and several fly-by-wire flights, future work can continue the build up to fully autonomous flight. The capability for autonomy already exists in the system and only needs to be demonstrated. Additionally, the SAR software can be run through more scenarios to further prove its utility. As full autonomy is reached, full NDVI maps of fields can be generated.

XI. Conclusions

This paper outlines the necessity for a dual-use UAS capable of performing Search and Rescue and Precision Agriculture missions. It delves into details of the design, component selection, construction, testing, and results of TIGRESS. This is a prototype UAS that has completed initial airworthiness checks and generated promising results. Future work is also discussed. It includes continuing flights to demonstrate fully autonomous operations, and further verifying the performance of the software suites

XII. Acknowledgments

NASA LaRC
Elizabeth Ward
Ken Dudley
Jeff Hill
Sam James
Truong Nguyen
Patricia Oneal
Patrick Quach
Garry Qualls
Kyle Smalling
Tom Strom
Brent Weathered
Brandon Basso
Kenny Deyerle
Nancy Holloway
Mike Logan
Kevin Vipavetz
Gary Wainwright
Jeff Werner

XIII. References

[1] http://www.uscg.mil/hq/cg5/cg534/SARfactsInfo/SAR%20Sum%20Stats%2064-14.pdf [Retrieved June 2014]

[2] https://www.uscg.mil/hq/cg5/cg534/manuals/COMDTINST%20M16130.2F.pdf [Retrieved June 2014]

[3] http://www.nrdc.org/media/2013/130827.asp [Retrieved June 2014]

[4] http://www.e-fliterc.com/ProdInfo/Files/EFLDeterminingPowerRequirement.pdf [Retrieved June 2014]

[5] Rabatel, G., Gorretta, N., Labbé S., “Getting NDVI Spectral Bands from a Single Standard RGB Digital Camera: A Methodological Approach”, *14th Conference of the Spanish Association for Artificial Intelligence*, La Laguna, Spain, 2011

[6] Weier, J. and Herring, D.m “Measuring Vegetation (NDVI & EVI)”, NASA Earth Observatory, 2000 [Retrieved from:earthobservatory.nasa.gov/Features/MeasuringVegetation/measuring_vegetation1.php
[cited August 1, 2014].

[7] Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Preibisch, S., Reuden, C., Saafeld, S., Schmid, B., Tinevez, J., White, D., Hartenstein, V., Eliceiri, K., Tomancak, P., Cardona, A., "Fiji: an open-source platform for biological-image analysis", *Nature Methods, Vol. 9, No. 7, 2012, p. 676-682.*

[8] Sokalski, Jan, and Breckon, Toby. Automatic Salient Object Detection in UAV Imagery. *25th International UAV Systems Conference*