Multi-UAS Path Planning for Non-Uniform Data Collection in Precision Agriculture

Patrick Nolan
Heron Systems, Inc.
2121 Eisenhower Avenue,
Suite 401
Alexandria, VA 22314
571-257-8403
pat.nolan@heronsystems.com

Derek A. Paley
University of Maryland, College Park
3150 Martin Hall
4298 Campus Dr.,
College Park, MD 20742
301-405-5757
dpaley@umd.edu

Kenneth Kroeger
Heron Systems, Inc.
2121 Eisenhower Avenue,
Suite 401
Alexandria, VA 22314
571-257-8403
ken.kroeger@heronsystems.com

Abstract— **This paper presents an augmented path-planning technique for unmanned aerial systems to generate focused trajectories about one or more areas of interest for non-uniform sensor data collection. The technique described in this paper uses a coordinate transformation that augments the work space with a temporary, virtual space in which existing path-planning and control algorithms can be used to provide uniform coverage. Transforming back to the original work space forces the planned trajectories to focus on regions of interest. We illustrate the application to precision farming, where regions of interest in a crop field correspond to stressed crop health. When collecting aerial survey data, we seek to have a higher density of sensor data in areas of interest (e.g., RGB images, multispectral images, etc.). The technique presented in this paper offers a method for concentrating sensor measurements around these regions of stressed crop health for one or more vehicles. In agricultural domains with multiple regions of interest, a Voronoi partitioning algorithm partitions the operating area into individual regions in which the augmented path-planning technique is applied. The path-planning in each region takes into account the resources available—i.e., vehicles with larger sensor footprints are assigned to larger regions and execute trajectories that are more broadly spread as compared to vehicles with smaller sensor footprints. Theoretical results are applied to commercial off-the-shelf unmanned systems, both in simulation and in a fully realized precision agriculture demonstration field experiment.**

TABLE OF CONTENTS

1. INTRODUCTION 1
2. PROBLEM FORMULATION......................................2
3. TRANSFORMATION TO AND FROM THE AUGMENTED PLANNING SPACE 3
4. DETERMINING VEHICLE FLIGHT PATHS 4
5. PARTITIONING SPACE BASED ON REGIONS OF INTEREST...5
6. APPLYING TO COTS VEHICLES 6
7. FLIGHT RESULTS..7
8. CONCLUSIONS 10
ACKNOWLEDGEMENTS 10
REFERENCES 10
BIOGRAPHY 11

1. INTRODUCTION

The emerging prevalence of unmanned aerial systems (UAS) for collecting a wide array of data in many different industries increases the need for practical methods of utilizing common aerial platforms. In industries where data is spread over a large geographic area or where an aerial perspective is the optimal data set, UAS are a popular choice. For example, UAS in the agriculture industry are deployed to fly over a field of interest and collect imagery, which is usually stitched together to form an orthomosaic image of the entire field at a single point in time. The orthomosaic can be processed to determine if and where there may be areas of poor crop health in the coverage area using techniques discussed by Saxena and Armstrong and others [1][2]. This approach gives the farmer an invaluable asset to their operations—a quick way to attain a snapshot of their field with a clear depiction of potential problem areas in order to take action sooner than previously possible.

The aerial perspective orthomosiac process is accurate to a certain degree, but the accuracy of the resulting orthomosiac is highly dependent on the image resolution and flight altitude when collecting input imagery [3]. The accuracy (as low as 1.25 m error over a 2.1 km^2 area) is such that using the resulting imagery, one can determine a farm's compliance with various conservation buffers and other agricultural conservation efforts in a region [4][5][6]. Farmers who prove compliance with these conservation efforts receive a number of tax benefits, and UAS can provide a faster, cheaper alternative to overhead satellite imagery or manned aircraft solutions. In addition to RGB orthomosaics, we use NDVI to gain valuable insights into the health of the field and estimate yield [7][8]. The NDVI values across an image have a strong

978-1-5090-1613-6/17/$31.00 ©2017 IEEE

correlation with the actual yield of the field—i.e., healthier crops exhibit a higher NDVI value and typically produce a higher yield. The opposite is true with crops that exhibit a lower NDVI value [9][10].

There are limitations, however, that come with the use of UAS. Common commercial off-the-shelf (COTS) unmanned systems, while affordable and readily available, typically struggle to cover the acreage occupied by an average farm. As an example, the average farm size in the United States is approximately 434 acres, and the average farm size in the state of Virginia is 180 acres [11]. Even the smaller Virginia farms are difficult to cover with one vehicle without numerous battery changes. Due to their endurance and range limitations, it becomes imperative to maximize their time aloft on the task at hand. In the agricultural case, the task is to collect aerial imagery that ensures coverage of areas of stressed crop health while also maximizing the coverage of the field under survey. Conventional coverage path-planning methods do not take into account regions of interest, and instead cover the entire space.

This paper presents a method to plan vehicle paths that focus on a region of interest defined by a single point, e.g., at the centroid of the region. The resulting path covers the entire space while also ensuring important regions are covered more densely. The farmer receives a processed aerial image set of their field that has identified potential areas of poor crop health. Further, the assessment accuracy is greater in the regions of interest, due to increased image overlap. The method presented in this paper is a specialization of the methods presented by Sydney and Paley for multivehicle coverage of a nonstationary spatiotemporal field [12].

In the case of multiple regions of interest in the same operating space, it is beneficial to use multiple UAS platforms to cover the entire space and to ensure the regions of interest are adequately covered. We propose to partition the operating space using the regions of interest as inputs to a Voronoi partitioning algorithm. A Voronoi partition is generated using a pre-specified set of points and, for each point, a region is found consisting of all points closer to that point than any other in the set [13]. Work has been done exploring load sharing and path planning using Voronoi tessellations specific to UAS applications. In military applications, we often don't want or need multiple UAS in the same airspace. Therefore, the operational space is divided into separate regions and each agent is tasked to stay within its own region [14]. Once we have individual regions for each aircraft, path planning in each individual Voronoi region can follow typical path-planning architectures [15].

The path-planning method described in this paper is applied such that each vehicle focuses its trajectories in a Voronoi partition defined by a point of interest, and the paths generated are simple lawnmower survey patterns. In an operational sense, farmers provide initial points of interest in their fields to be used as inputs to the partitioning and path-planning algorithms.

The contribution of this paper is a path-planning method that generates vehicle trajectories aimed at tackling real-world data-sampling problems in the agricultural industry. Results are illustrated in simulation and onboard actual COTS aircraft in a fully realized flight operation. A graphical user interface (GUI) was developed to generate vehicle trajectories, upload them to the aircraft, and perform basic vehicle commands (i.e., takeoff, land, perform mission, and return to launch). With COTS aircraft and RGB cameras, we show that the algorithm presented outperforms typical survey patterns when detecting image keypoints, matched keypoints, and 3D points observed in overlapping images.

The paper outline is as follows. Section 2 formulates the path-planning problem, focusing on agricultural applications. Section 3 outlines the overall procedure and equations used to transform the space to and from the planning domain. Section 4 briefly describes the method used to determine the flight paths for the UAS. Section 5 presents a partitioning scheme applicable to multi-vehicle operations consisting of multiple regions of interest. Section 6 discusses implementation onto COTS UAS platforms and subsequent flight operations carried out over a representative farm field. Section 7 summarizes the paper and ongoing work.

2. Problem Formulation

Agricultural operations are typically broken down by individual fields—rarely does an action apply to every acre of a farm. An individual field is defined by three or more latitude and longitude coordinate pairs converted to *(x, y)* in meters to form a convex polygon. We convert the *(x, y)* coordinate pairs from Cartesian coordinates to the complex plane, which is useful when performing the transformations described in Section 3 to and from the planning space.

Points of interest are defined by a single latitude and longitude pair. For this work, this coordinate pair is located roughly at the centroid of an area of interest, which ensures the resulting path covers the full area of interest. In the precision agriculture applications discussed in this paper, the points of interest do not need to be exactly positioned because they represent a region of potentially poor crop health, which is generally not well localized.

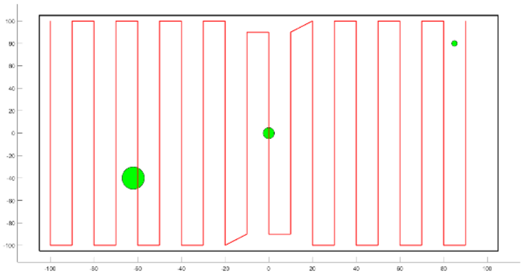


Figure 1: An example of a simple lawnmower search pattern with points of interest denoted in green. Not all of the regions are adequately sampled by the flight pattern.

With the field boundary and points of interest now defined, we proceed to plan a path to adequately cover the entire space. However, typical survey patterns potentially miss some important data or may not collect sufficient data about problem areas in the operating space. Figure 1 illustrates what a typical lawnmower flight pattern looks like in a rectangular field. Notice how the flight pattern fails to concentrate measurements on regions of interest. One solution is to create a uniform flight pattern denser than before, as depicted in Figure 2. A dense survey ensures that all points are adequately sampled, but is not always possible due to the endurance limitations of various COTS UAS.

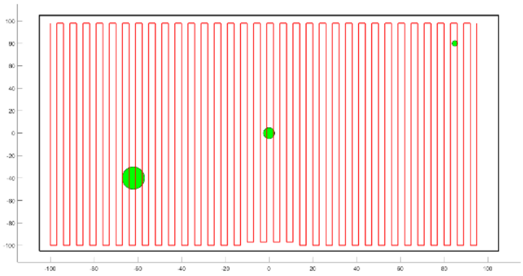


Figure 2: A dense survey pattern captures all of the points of interest, but is not necessarily feasible due to energy and/or time constraints.

The following assumptions are made in this work. First, all boundaries are convex in shape. This assumption simplifies the path-planning portion of the work, and helps with the real-world agricultural implementation. Also, the COTS vehicles are multi-rotor platforms. This assumption allows greater flexibility in path-planning, as multi-rotors can make sharp turns and reach a wide variety of spaces that fixed-wing platforms cannot. However, the methods described in this paper can also be applied to fixed-wing platforms, with the caveat that the resulting trajectories would have to be smoothed and additional considerations taken to ensure a safe airspace.

3. Transformation to and from the Augmented Planning Space

The ideal survey path balances endurance and sample density. Such a flight pattern may focus on a point of interest while remaining spread out in regions of less interest. To define such a pattern, consider the following procedure:

1. Define (complex) boundary vertices;
2. Transform the boundary into the augmented space, i.e., the *R*-domain;
3. Define a planning grid in the *R*-domain;
4. Define a uniform survey path from the planning grid; and
5. Transform the survey path back to the original space, i.e., the *r*-domain.

The set of points that define the boundary in the complex plane can also be expressed in polar coordinates:

$$Z = x + iy \tag{1}$$

$$(x, y) = (|Z|\cos\theta, |Z|\sin\theta) \tag{2}$$

$$|Z| = \sqrt{x^2 + y^2}\ ;\ \theta = \arctan\left(\frac{y}{x}\right) \tag{3}$$

Next, transform the boundary points into an augmented space such that angles are preserved, and the distance from the point of interest (POI) is decreased. The augmented space, i.e. the R-domain, is defined by the following transformation:

$$|R| = m|Z - POI|^p \tag{4}$$

$$\angle R = \angle(Z - POI) \tag{5}$$

$$R = |R|\cos(\angle R) + |R|\sin(\angle R)\, i, \tag{6}$$

where $|R|$ is the magnitude of the point in the augmented space and $\angle R$ is the argument of the point in the augmented space. Notice that $|R|$ and $\angle R$ are calculated with respect to POI. If the POI is the origin, then space is warped about the origin; otherwise shift the transformation center to the POI.

The constants m and p are factors that drive the amount of augmentation that occurs. The constant p is defined by the user to vary the amount of warping about the point of interest. The smaller the value, the more the resulting path will be concentrated about the POI. The value must be positive, non-zero, and less than one to act as a sink towards the POI. A value greater than one acts as a repulsive force away from the POI. The p constant has a nonlinear effect on path distortion, in that it has a larger effect the closer the path is to the POI. The constant $m = 10^{(1-p)}$ has a uniform effect on the path contraction, i.e., the effect is the same over the entire planning space.

At this point, we have the augmented space in which to plan a vehicle path. Determining the path will be discussed in more detail in Section 4. In short, the planned path is made up of a set of points in the complex plane that form a lawnmower-style path. The final step is to transform the augmented boundary and the planned path from the augmented space back to the original space, i.e., the *r*-domain. Note that we convert the waypoints in the augmented space, denoted by WP, back to the original complex space, denoted by wp.

$$|r| = \left(\frac{1}{m}\right)|R|^{\left(\frac{1}{p}\right)} + Re(POI) \qquad (7)$$

$$\angle r = \angle(R) + Imag(POI) \qquad (8)$$

$$r = |r|\cos(\angle r) + |r|\sin(\angle r)\, i \qquad (9)$$

$$|wp| = \left(\frac{1}{m}\right)|WP|^{\left(\frac{1}{p}\right)} + Re(POI) \qquad (10)$$

$$\angle wp = \angle(WP) + Imag(POI) \qquad (11)$$

$$wp = |wp|\cos(\angle wp) + |wp|\sin(\angle wp)\, i \qquad (12)$$

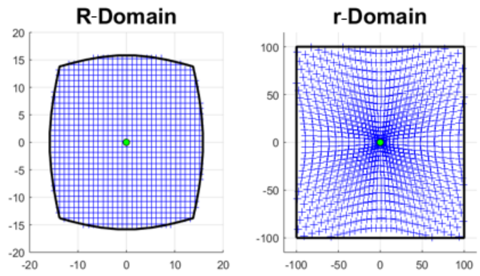


Figure 3: Transformation to and from the *R*-domain, the augmented planning space.

Figure 3 shows the results of the above process, not including the vehicle path. Notice the difference in axis scales, which is a result of the scaling and warping by the m and p values. On the left, note that the boundary is warped outward about the origin. The figure on the right shows the original space: observe that the grid is concentrated towards the point of interest at the origin. Figure 4 shows the same process with a point of interest other than the origin.

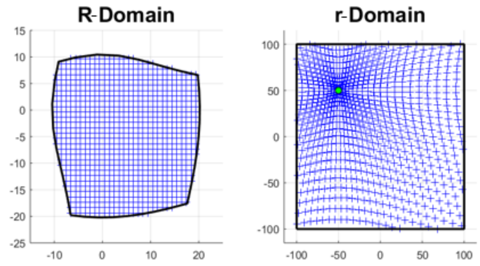


Figure 4: The transformation applied to a point of interest other than the origin.

4. Determining Vehicle Flight Paths

Next we determine the vehicle trajectories that cover the augmented space. Many coverage algorithms exist [16][17], but the one most typically implemented for COTS UAS is the so-called lawnmower pattern. In this application, we implement a lawnmower pattern based on grid points inside the R-domain boundary. The spacing between the grid points is determined by the sensor footprint of each aircraft—the larger the sensor footprint, the more spread out the grid will be and, subsequently, the more spread out the flight pattern is.

To define the grid in the R-domain, define a minimum bounding box to encompass the augmented boundary. In that minimum bounding box, define a mesh grid of points with the spacing as defined above and find the points in that mesh contained in the polygon defined by the transformed boundary points in the R-domain. Figure 5 illustrates this step.

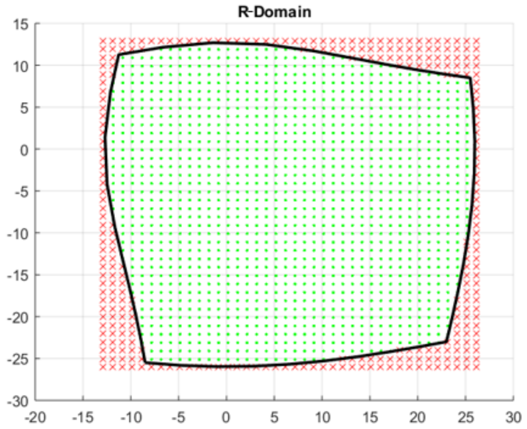


Figure 5: A grid in the minimum bounding box containing the augmented boundary. The points inside the polygon box represent the path-planning grid.

Using the grid points as proxies for vehicle waypoints defines a lawnmower pattern. Connect the points in order to create the path. Notice how the resulting path is focused on the point of interest. The point of interest acts as a sink, pulling the flight pattern inward.

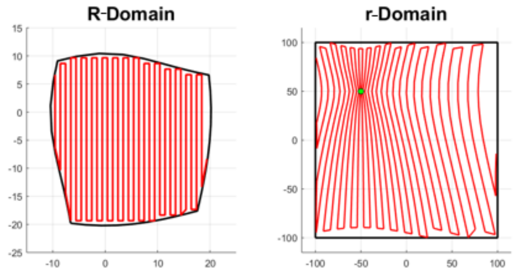


Figure 6: Planned paths in the *R*-domain (left) and the *r*-domain (right). Note that the resulting path in the *r*-domain is concentrated about the point of interest while still covering the entire *r*-domain.

5. Partitioning Space Based on Regions of Interest

Now that we have a method for concentrating a flight pattern about a point of interest, we consider the multi-vehicle applications of this approach. The use case envisioned here is when the user has a boundary that encompasses multiple regions of interest. For example, each region of interest represents a region of potentially poor crop health, defined as a single point of interest.

With the boundary points and the points of interest defined, we partition the operating space into smaller spaces, each corresponding to a point of interest assigned to a vehicle. For this, we implemented a modified Voronoi partitioning algorithm in Python [18]. A Voronoi diagram partitions a plane into regions based on the distance to a set of predefined points. A typical Voronoi diagram has partitions with non-closed boundaries (i.e., boundary edges that go to infinity). The algorithm used here finds the intersections of these infinite lines with the user-defined boundary to close the Voronoi cell into a convex polygon. If the user-defined boundary is not rectangular, a minimum bounding box is used instead. Points of interest are defined by the user and input to the Voronoi algorithm. The output of the resulting Python script for three points of interest is shown in Figure 7.

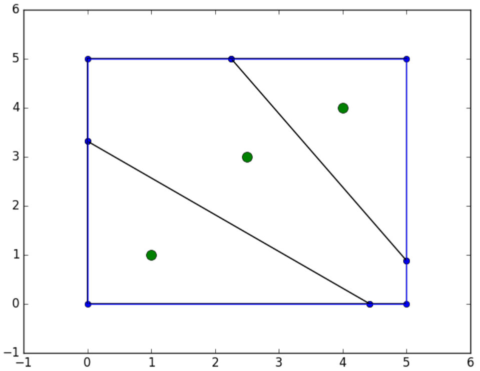


Figure 7: Results of the Voronoi algorithm for three points of interest.

The Voronoi cells defined in Figure 7 are convex polygons that are used as inputs to the path-planning algorithm. Using the same points of interest and boundaries yields the results shown below in Figure 8. These results assume each vehicle has the same sensor footprint, and therefore the planned paths have the same spacing between legs.

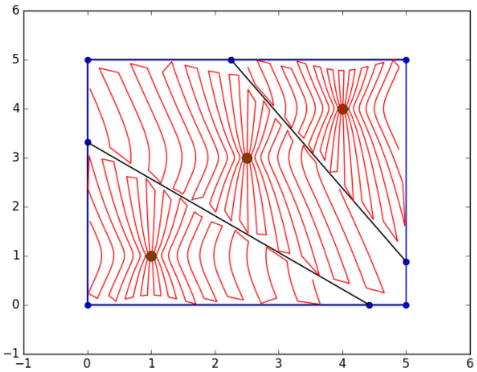


Figure 8: Applying the path-planning method to each Voronoi cell.

6. APPLYING TO COTS VEHICLES

The overall intention of this work is to tackle the real-world problem of determining more useful flight patterns for COTS UAS in precision agriculture applications. The methods defined in the previous sections were integrated with several common UAS on the market at the time of publication. The targeted aircraft are those that implement the open-source MAVLink Micro Air Vehicle Communication protocol, such as the 3DR Solo or other aircraft driven by the Pixhawk autopilot. (At the time of this writing, a Python API labeled DroneKit [19], has been developed by 3DR to interface with MAVLink enabled aircraft.)

We ported the MATLAB path-planning code into Python so that the software is run entirely in the same environment. In addition, implementing the source code in Python gives the flexibility to interface with a wide array of other software languages and applications. With the entire source code in Python, interfacing with the DroneKit Python API is seamless. The API allows us to connect to a MAVLink-enabled aircraft and upload a mission to the aircraft. Using the Python scripts, we save the planned paths in a format readable by the aircraft. Once the path is converted to a usable format, we connect and upload to the aircraft.

A Graphical User Interface (GUI) was developed to help visualize the operating space and planned paths for each aircraft. Leaflet [20], an open-source mapping toolbox, was used to visualize the fields and flights. The GUI, depicted in Figure 9, allows a user to quickly change the values for the *p* and *m* constants, as well as the grid spacing. For the GUI in this work, the grid spacing is the number of passes that the aircraft will take according to the size of the sensor footprint.

The GUI also allows the user to quickly change the path direction from North-South to East-West. It uses the Python software developed to generate the paths for each vehicle, with two modifications made with

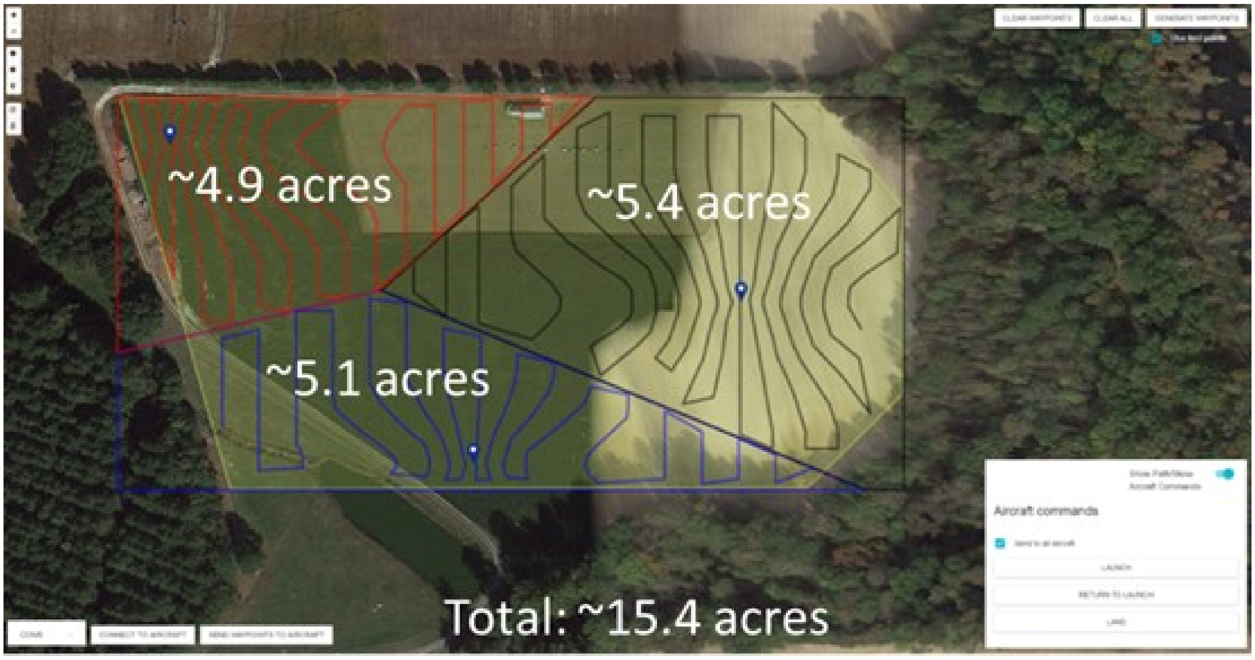


Figure 9: Vehicle paths generated for three MAVLink-enabled COTS UAS. The hot spots are displayed with a marker icon, and the three paths are warped towards each hot spot.

real-world flight operations in mind: first, as we move away from a hot spot, the sink effect it has on the path is less pronounced because we only care about augmenting the path when the camera footprint is able to capture the region of interest. This results in a decaying effect of the pull by the hot spot on the vehicle path. The second modification is to account for non-rectangular boundaries. With a non-rectangular boundary, we find a minimum bounding box to plan the vehicle paths and discard waypoints that do not lie in the original non-rectangular boundary.

The methods developed in this work were demonstrated onboard a fully realized aircraft implementation. Using multiple 3DR Solo platforms, we were able to plan a path for each aircraft using the above methods, upload the paths to each vehicle, and collect aerial imagery over a plot of farmland in the Tappahannock, Virginia area. The vehicle paths for an October, 2016 flight are shown in the GUI screenshot in Figure 9. All three vehicles to completed their portion of the overall survey in approximately 20 minutes. The vehicle denoted by the red path covered approximately 4.9 acres, whereas the vehicles denoted by the blue and black paths covered approximately 5.1 and 5.4 acres, respectively. In total, the imagery collected by the three aircraft covered approximately 15.4 acres.

7. Flight Results

The vehicle paths shown in red, blue, and black in Figure 9 were loaded onto three 3DR Solo quadcopters and flown autonomously with no pilot input. Note that the red, blue, and black boundaries combine to form the minimum bounding box of the non-rectangular boundary shown in yellow. The paths are first planned using the hot spots and the minimum bounding box, then the paths are trimmed of any waypoints not contained in the non-rectangular boundary of interest. Also note that the survey pattern legs farthest from the hot spot are straight instead of pulled towards the hot spot. The resulting stitched RGB orthomosaic is displayed below.

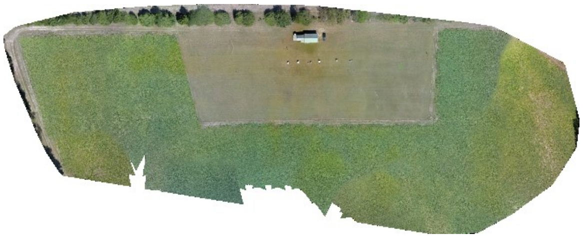


Figure 10: RGB orthomosaic resulting from multi-vehicle flight operations.

The orthomosaics for this work were created using the Pix4D image stitching cloud software using the geotagged images resulting from each survey pattern. For each orthomosaic, we look at various key statistics of the stitching process and resulting calibrations. Specifically, we look at the mean and median number of keypoints per image, the median matched keypoints per image, and the number of 3D points observed in two overlapping images. We also look at the average point density (per m^3) of the resulting point cloud as a measure of point cloud accuracy. The stitching process metrics for the orthomosaic in Figure 9 outperform almost every corresponding metric for an orthomosaic generated from a typical survey pattern over the same area with comparable flight times. Figure 10 shows the resulting metrics from the stitching process for each flight.

A higher number of keypoints per image typically means that we can match a higher number of keypoints between multiple images. The stitching process is more accurate with more matching features between images. Therefore, the higher the number of matching keypoints, the higher the accuracy will be for the resulting orthomosaic. Also, a higher number of matched keypoints typically results in a higher number of 3D points observed in overlapping images, which improves the accuracy of a point cloud generated from the same image data.

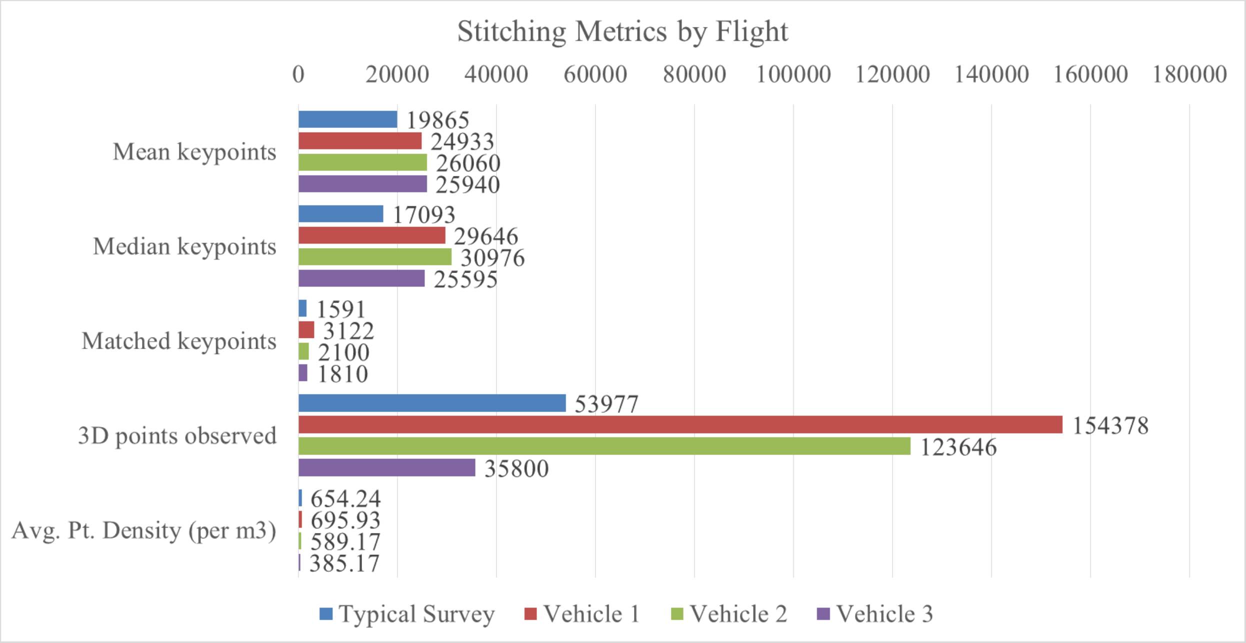


Figure 11: The augmented survey typically collects more keypoints, matches more keypoints, and observes more 3D points than a typical survey pattern in the same region of interest.

The same analysis was performed for a rectangular boundary with a single aircraft. We flew a typical survey pattern in a small area followed by an augmented survey pattern in the same area, shown in Figure 12. The results of those flights are detailed in Figure 13. Again, we see that the augmented pattern has a higher mean and median amount of keypoints detected per image. We also see that the number of matched keypoints per image is increased, and in this case the number of 3D points is also increased. These metrics apply for the entire flight pattern, so the overall accuracy over the entire orthomosaic will be improved using the augmented survey pattern. The overall improvement is driven by a larger improvement in the region of interest compared to a typical survey pattern. For this testing, the imagery collected by the vehicle covered approximately 1.16 acres.

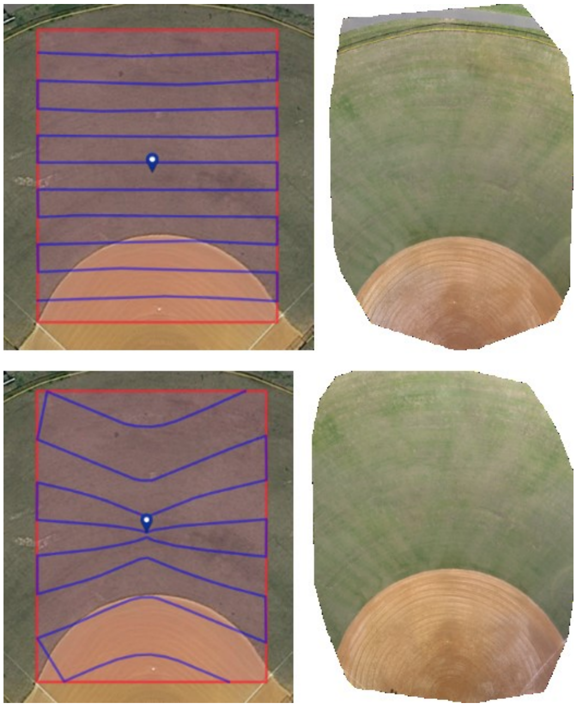


Figure 12: (TOP) Typical survey pattern with the resulting orthomosaic. (BOTTOM) Augmented flight pattern with the resulting orthomosaic.

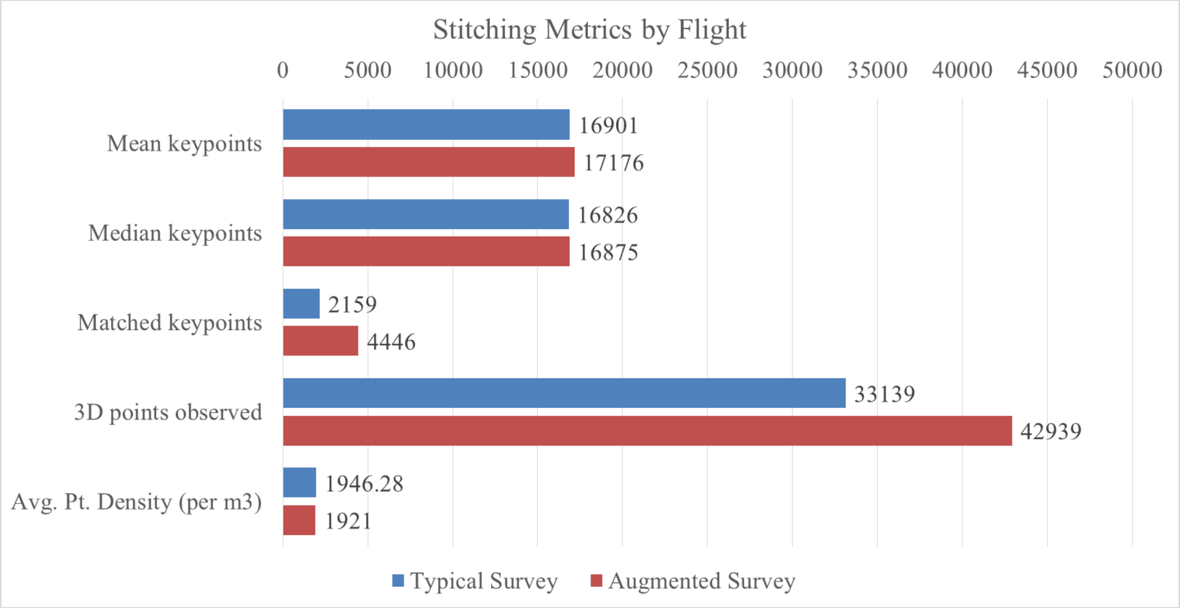


Figure 13: The augmented survey slightly outperforms the typical survey pattern on a smaller flight area.

This particular implementation was not without its limitations. Unfortunately, we could not guarantee that the images collected over the resulting flight plans would be of sufficient quality to be used in the image stitching process. Examining Figures 9 and 10, we see that portions of each flight trajectory are curved. What this means is that the vehicle will be maneuvering throughout that portion of the trajectory. In this test, the vehicles were moving slow enough so that they only had to yaw to initiate a turn. However, even with a simple yaw, there is some variation in the roll and pitch of the aircraft which can skew some of the images. This occurs mostly at the edges of the flight patterns when the aircraft is turning around as the yaw is larger in magnitude. The result is that the images collected on the edge of the space often do not provide sufficient enough data for the orthomosaic process and we see a clipping or tearing effect (see the bottom of Figure 10).

Camera field of view (FOV) is an important aspect of flight planning to ensure that the imagery collected will have sufficient overlap for orthomosaic generation. In this work, to be consistent, we flew all flight profiles from an altitude of 19 meters (or 62.34 feet) and used a GoPro HERO4 Black camera [21]. The Hero 4 Black flown at 19 meters gives a FOV of approximately 39 m x 29.3 m. In any flight profile, we want to ensure that there is sufficient overlap in enough of the images to give us better accuracy when generating an orthomosaic [22]. For the purposes of this work, we aimed to have roughly 60%-75% overlap between legs of a normal survey pattern in both the horizontal and vertical dimensions.

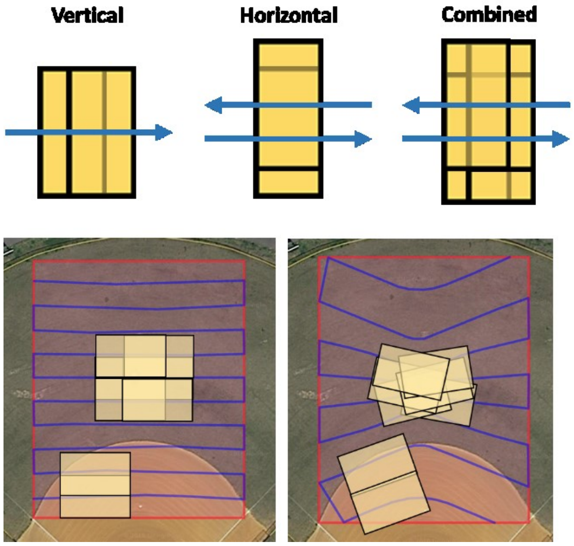


Figure 14: (TOP) Veritcal and Horizontal overlap defined, with direction of flight shown in blue. (BOTTOM) Image overlap about the point of interest increases due to the augmented flight pattern. A drawback is the loss of sufficient overlap in the outer regions of the flight pattern.

In Figure 14 (TOP), we define vertical overlap as the overlap between subsequent images in the direction of travel for the UAS. Horizontal overlap is the overlap between images in adjacent legs in a typical survey pattern. Figure 14 (BOTTOM) shows an example FOV overlaid on portions of the flight patterns from Figure 12. We can see that the augmented flight pattern with the same number of images in a region sees more image overlap over the POI than a typical survey pattern. One drawback of this particular implementation is shown in the bottom left-hand region of Figure 14, where we lose some horizontal overlap between survey legs in portions of the flight pattern that become spread out due to the effect of the POI.

8. Conclusions

We present a path-planning method tailored for unmanned aerial systems that generates a vehicle path focused about a region of interest. A coordinate transformation is used to determine an augmented planning space that can be uniformly sampled by existing planning techniques. When transformed back into the original domain, the vehicle paths tend to be concentrated about the region of interest, resulting in a denser sampling by sensors onboard the vehicle. In a multi-vehicle operation, the operating space is partitioned based on the regions of interest and available vehicle resources by a Voronoi partitioning scheme. The methods described in this paper are implemented into a precision agriculture operating environment consisting of several COTS UAS. We show that the augmented algorithm can provide more keypoints per image and subsequently more matched keypoints than a typical survey pattern, which result in orthomosaics with higher accuracy. We also show that more keypoints also typically result in more 3D points observed in overlapping images, and ultimately provide more accurate point clouds, particularly in the region of interest. Concentrating the flight patterns towards a region of interest improves the orthomosaic accuracy in the region of interest; preserving a typical survey pattern as we move further from a region of interest helps keep the flight times comparable to a typical survey.

In ongoing work, we are exploring the capabilities of other path-planning algorithms and sensors applied to the agricultural domain. Optimizing the planned vehicle paths in response to available resources and operating space geometry is the subject of future work, along with methods to reduce image artifacts resulting from dramatic shifts in vehicle orientation. Future work will also explore modifications to the algorithm presented to help mitigate the loss of horizontal image overlap in portions of the flight path that are further away from the point of interest.

Acknowledgements

The authors thank Bob Waring for the use of his farm fields for final demonstration purposes.

This work was supported in part by the National Aeronautics and Space Administration under the NNX16CL24C Phase 2 SBIR contract.

Flight operations for this work were carried out by certified remote pilots (Patrick Nolan—Cert. No.: 3913856; Kenneth Kroeger—Cert. No.: 3910113; Joseph Sochurek—Cert. No.: 3918443).

References

[1] Saxena, L., & Armstrong, L. (2014). A survey of image processing techniques for agriculture. Proceedings of Asian Federation for Information Technology in Agriculture. (pp. 401-413). Perth, W.A. Australian Society of Information and Communication Technologies in Agriculture

[2] “Precision Agriculture and Hyperspectral Sensors: Monitoring Against Drought, Disease, and Nutrient Stress.” *Precision Agriculture & Hyperspectral Sensors*. Surface Optics Corp, n.d. Web. 18 Aug. 2016. <https://surfaceoptics.com/applications/precision-agriculture-hyperspectral-sensors/>.

[3] Küng, Olivier, et al. "The accuracy of automatic photogrammetric techniques on ultra-light UAV imagery." *UAV-g 2011-Unmanned Aerial Vehicle in Geomatics*. No. EPFL-CONF-168806. 2011.

[4] Anton, Wilma Rose Q., George Deltas, and Madhu Khanna. "Incentives for environmental self-regulation and implications for environmental performance." *Journal of environmental economics and management* 48.1 (2004): 632-654.

[5] "Verification Program." (n.d.): n. pag. Pennsylvania Department of Environmental Protection. Pennsylvania Department of Environmental Protection, 16 Nov. 2016. Web. 4 Oct. 2016. <https://www.dep.state.pa.us/river/iwo/chesbay/docs/PADEPBMPVerificationProgramQAPPAddendumNov16.pdf>.

[6] Virginia, Commonwealth Of. (n.d.): n. pag. Virginia Agricultural Cost Share Manual. Virginia Department of Conservation and Recreation, May 2016. Web. 26 Sept. 2016.

[7] Hayes, J. T., et al. "YIELD: A numerical crop yield model of irrigated and rainfed agriculture." *Publ. Climatol.;(United States)* 35.2 (1982).

[8] "Measuring Vegetation (NDVI & EVI)." NASA. NASA, n.d. Web. 26 Sept. 2016. <http://earthobservatory.nasa.gov/Features/MeasuringVegetation/measuring_vegetation_2.php>.

[9] Prasad, Anup K., et al. "Crop yield estimation model for Iowa using remote sensing and surface parameters." *International Journal of Applied Earth Observation and Geoinformation* 8.1 (2006): 26-33.

[10] Quarmby, N. A., et al. "The use of multi-temporal NDVI measurements from AVHRR data for crop yield estimation and prediction." *International Journal of Remote Sensing* 14.2 (1993): 199-210.

[11] "USDA – NASS, Census Of Agriculture – 2012 Census Volume 1, Chapter 2: State Level Data". *Agcensus.usda.gov*. N.p., 2016. Web. 18 Aug. 2016.

[12] N. Sydney and D. A. Paley. Multivehicle coverage control for nonstationary spatiotemporal fields. Automatica, 50(5):1381–1390, 2014.

[13] Aurenhammer, Franz. "Voronoi diagrams—a survey of a fundamental geometric data structure." ACM Computing Surveys (CSUR) 23.3 (1991): 345-405.

[14] Pavone, Marco, Ketan Savla, and Emilio Frazzoli. "Sharing the load." *IEEE robotics & automation magazine* 16.2 (2009): 52-61.

[15] Maddula, Theju, Ali A. Minai, and Marios M. Polycarpou. "Multi-Target assignment and path planning for groups of UAVs." *Recent Developments in Cooperative Control and Optimization*. Springer US, 2004. 261-272.

[16] Arkin, Esther M., Sandor P. Fekete, and Joseph S. B. Mitchell. Approximation Algorithms For Lawn Mowing And Milling. 1st ed. 1995. Web. 18 Aug. 2016.

[17] Galceran, Enric, and Marc Carreras. "A survey on coverage path planning for robotics." *Robotics and Autonomous Systems* 61.12 (2013): 1258-1276.

[18] Flabetvibes. (2015, November 9). Getting a bounded polygon coordinates from Voronoi cells [Msg 2]. Message posted to http://stackoverflow.com/questions/28665491/getting-a-bounded-polygon-coordinates-from-voronoi-cells

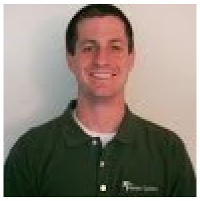
[19] "Welcome To Dronekit-Python'S Documentation!". Python.dronekit.io. N.p., 2016. Web. 18 Aug. 2016.

[20] Agafonkin, Vladimir. "Leaflet - an Open-source JavaScript Library for Interactive Maps." Leaflet Dev Blog Atom. N.p., 2015. Web. 05 Oct. 2016. <http://leafletjs.com/>.

[21] "GoPro Hero4 Black Specs." GoPro HERO4 Black Specs. CNET, n.d. Web. 20 Dec. 2016. <https://www.cnet.com/products/gopro-hero4-black/specs/>.

[22] "Step 1. Before Starting a Project 1. Designing the Image Acquisition Plan A. Selecting the Image Acquisition Plan Type." Support. Pix4d, 15 Dec. 2016. Web. 20 Dec. 2016. <https://support.pix4d.com/hc/en-us/articles/202557459-Step-1-Before-Starting-a-Project-1-Designing-the-Image-Acquisition-Plan-a-Selecting-the-Image-Acquisition-Plan-Type#gsc.tab=0>.

BIOGRAPHY



***Patrick Nolan** received a B.S. in Aerospace Engineering from the University of Maryland, College Park in 2012. He has been with Heron Systems, Inc. for approximately 4 years. He has been a part of numerous Small Business Innovative Research (SBIR) projects and several IR&D projects pertaining to practical applications of unmanned systems. His research interests include multi-vehicle planning and control for use in real world scenarios, such as precision agriculture, and civil/mining applications.*

***Derek A. Paley** is the Willis H. Young Jr. Associate Professor of Aerospace Engineering Education in the Department of Aerospace Engineering and the Institute for Systems Research at the University of Maryland. Paley received the B.S. degree in Applied Physics from Yale University in 1997 and the Ph.D. degree in Mechanical and Aerospace Engineering from Princeton University in 2007. He received the National Science Foundation CAREER award in 2010, the Presidential Early Career Award for Scientists and Engineers in 2012, the University of Maryland E. Robert Kent Teaching Award for Junior Faculty in*

2014, and the AIAA National Capital Section Engineer of the Year in 2015. Paley's research interests are in the area of dynamics and control, including cooperative control of autonomous vehicles, adaptive sampling with mobile networks, and spatial modeling of biological groups.

Kenneth Kroeger received his B.S. and M.S. degrees in Mechanical Engineering from Virginia Polytechnic Institute & State University in 2010 and 2013 respectively and is actively pursuing his Doctoral degree. He is the Primary Investigator of a NASA Phase 2 SBIR program supporting this work titled A Modular Swarm Optimization Framework Enabling Multi-Vehicle Cooperative Path Planning. Kroeger's research interests include collaborative control and path-planning for unmanned systems, enabling real-time computer vision technologies, and electrical system architecture/design.