

**PHOTOGRAMMETRY AND LIDAR FROM THE UNMANNED AERIAL PLATFORM FOR  
MEASURING THE PHYSICAL STRUCTURE OF FORESTS**

By

**H. ANDREW LASSITER**

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL  
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT  
OF THE REQUIREMENTS FOR THE DEGREE OF  
**DOCTOR OF PHILOSOPHY**

UNIVERSITY OF FLORIDA

2018

© 2018 H. Andrew Lassiter

To Faraday

## ACKNOWLEDGMENTS

My journey to this highest of academic achievement has been anything but a solitary one. I am especially grateful for the support of my family, especially my grandmother, Wilma, my father, Jeff, and my brother, Adam. I do not take on this pursuit, much less make it to Gainesville, without their unwavering support.

This journey may not be taken alone, but the feelings of isolation and uncertainty still find host in academic minds. The camaraderie and companionship of my friends, many of whom are fellow graduate students, helped me find my footing and stay dedicated to taking care, both of myself and in my work, in those times when I doubted myself. I cannot believe how fortunate I am to say the following: there are too many of you to name here.

I am grateful to work alongside collegial and supportive fellow researchers, especially John Roberts, Ruoyu Wang, Travis Whitley, Luiz Felipe Ramalho de Oliveira, and Iván Raigosa García. Your candor and cooperation these past four years has made my time at the University fulfilling and enriching. I am a better scholar and colleague for working alongside you. This dissertation is 34 pages longer and at least 22% better because of you. We have done great work together, and I would be so lucky as to encounter colleagues of your caliber in my next chapter.

I have heard so many nightmare stories from my fellow graduate students about collecting data; these stories are so rampant that they are a sort of currency among researchers. I am happy to say that, in this sense, I am quite poor. Collecting data for my research has always been, dare I say it, easy. Thank you to Ordway Swisher Biological Station and Austin Cary Memorial Forest for providing access to so many trees.

The UFUAS Research Program has provided me an amazing opportunity to learn so much about the UAS platform while working alongside supportive and knowledgeable people. Thank you to Matthew Burgess, Ray Carthy, Ben Wilkinson, Peter Ifju, Scot Smith, Travis

Whitley, Chad Tripp, and Andrew Ortega. Working with you all the past two years has been a pleasure.

The support I have received in all matters from the School of Forest Resources and Conservation is incomparable. In my years studying at the University, I have always had access to opportunities and assistance thanks to SFRC faculty and staff. I have encountered an openness and willingness to help here that I hope to find again in my future endeavors.

A special thanks to my committee members Grenville Barnes, Amr Abd-Elrahman, Jason Vogel, and Peter Ifju. Your commitment to my scholarship is evident in our interactions, from examining my research and brainstorming new avenues of inquiry to working alongside you in the field. I should have asked you all a thousand more questions during my time here; curse my habit of working in solitude.

To my advisor, Ben Wilkinson, I owe a tremendous debt of gratitude. In the past few months I have rifled through data and prose spanning my time as a graduate student, and I could hardly recognize the previous version of my researching self. I have grown so much as a scholar under your guidance and tutelage. As recently as one year ago I did not believe that I was “Ph.D. material,” an all-too-common ailment among those who take to this journey. Through all that I have learned and through all the opportunities you have presented to me, I can say today that I no longer feel this way. I am ready to take this next step in my career in research and education. Thank you for everything.

## TABLE OF CONTENTS

	<u>page</u>
ACKNOWLEDGMENTS .....	4
LIST OF FIGURES .....	10
LIST OF ABBREVIATIONS.....	13
ABSTRACT.....	15
CHAPTER	
1    INTRODUCTION .....	17
UAS Lidar.....	17
UAS Lidar Payload Development .....	18
El-Sheemy Georeferencing Equation.....	19
Cubic Spline Interpolation of Navigation Data.....	19
UAS Photogrammetry .....	21
Forest Mensuration from Point Clouds.....	22
2    MISSION PLANNING FOR LOW-ALTITUDE AIRBORNE LASER SCANNING WITH THE VELODYNE VLP-16 .....	27
VLP-16 Configuration.....	28
Analytical Characterization of VLP-16 Scan Pattern.....	29
Point Density Function .....	29
Optimal Separation of Flight Lines .....	30
Gap Equation .....	30
Simulation of the VLP-16 Scan Pattern.....	31
Input and Output.....	31
Spatial and Error Analysis of Simulated Data.....	32
Results and Discussion .....	34
Single Strip .....	34
Overlapping Strips.....	35
Rotation Rate and Gaps .....	36
Yaw.....	36
Spatial and Error Analysis.....	37
3    OBLIQUE UAS PHOTOGRAHMTRY IN FORESTED SCENES .....	53
Introduction.....	53
Automated 3D Reconstruction of the Forested Scene.....	53
Feature Matching in a Forested Scene.....	54
Dynamic, Close-range Photogrammetry .....	55
Methods .....	57

Study Areas .....	57
Mission Simulation.....	58
Mission Planning .....	60
Reconstruction Analysis .....	62
<b>Results.....</b>	<b>63</b>
Mission Simulation and Planning.....	63
Quantitative Checkpoint Analysis .....	64
Qualitative CHM Analysis .....	64
<b>Application: Individual Tree Detection (ITD).....</b>	<b>65</b>
Methods .....	65
Results .....	66
<b>4 A STEM-BASED APPROACH TO SEMI-AUTOMATED ESTIMATION OF PINE MORPHOLOGY FROM LOW-ALTITUDE 3D MAPPING.....</b>	<b>79</b>
Introduction.....	79
Aerial Photogrammetry in Forestry .....	79
Automated Forest Mensuration from Point Clouds.....	81
Canopy-Based Versus Stem-Based Individual Tree Detection.....	83
Methods .....	83
Data Collection and Processing.....	83
Stem Detection Algorithm.....	86
Diameter at Breast Height Estimation Algorithm .....	87
Height Estimation Algorithm .....	89
Equivalence Testing .....	89
Results.....	90
Terrestrial Data, Preliminary Study.....	90
Dense Matching, ACMF Site .....	91
Lidar, Flatwoods Site.....	92
<b>5 CONCLUSION.....</b>	<b>105</b>
Recommendations for UAS 3D Remote Sensing .....	105
Forest Mensuration from UAS-borne Point Clouds .....	106
Current State of the Art.....	108
<b>APPENDIX</b>	
<b>A VLP-16 CHARACTERIZATION EQUATIONS .....</b>	<b>110</b>
Point Density Probability Density Function .....	110
Optimal Separation of Parallel Flight Lines .....	112
Possible Positions of Coverage Gaps .....	113
Generalizing the Equations for Yaw.....	114
Point density probability function .....	114
Optimal separation of flight lines .....	115
Gap equation.....	115

B	VLP-16 SIMULATION SOFTWARE .....	117
	Line-Plane Intersection.....	117
	Assumptions .....	118
	Optimization .....	118
	Azimuth bounds.....	119
	Time Bounds .....	119
C	DERIVATION OF OBLIQUE UAS PHOTOGRAMMETRY EQUATIONS .....	124
	Air Base and Air Width.....	124
	Ground Sample Distance .....	125
D	SIMULATING IMAGES IN MATLAB .....	127
E	CONVERSION OF ROTATION DATA BETWEEN ENU AND NED FRAMES .....	129
	Rotation Matrix from Yaw-Pitch-Roll, Xsens Definition .....	130
	Rotation from Yaw-Pitch-Roll, Aviation Convention.....	131
	LIST OF REFERENCES .....	133
	BIOGRAPHICAL SKETCH .....	138

## LIST OF TABLES

<u>Table</u>		<u>page</u>
2-1 Gap equation results, rotation rate 5 Hz.....	51	
2-2 Gap equation results, rotation rate 10 Hz.....	51	
2-3 Gap equation results, rotation rate 20 Hz.....	52	
3-1 PhotoScan processing details.....	74	
3-2 Processing specifications .....	74	
3-3 Checkpoint error results .....	74	
3-4 Checkpoint analysis, <i>F</i> -test results .....	75	
3-5 Checkpoint analysis, Welch's <i>t</i> -test results .....	75	
3-6 Summary statistics for the canopy height models.....	75	
3-7 Individual tree detection results from lidar CHM.....	76	
3-8 Individual tree detection results from nadir CHM.....	77	
3-9 Individual tree detection results from oblique CHM .....	78	
4-1 Stem detection summary.....	104	
4-2 DBH estimation summary.....	104	
4-3 Height estimation summary .....	104	

## LIST OF FIGURES

<u>Figure</u>		<u>page</u>
1-1	Multimodal corner cube target.....	24
1-2	Flowchart of lidar data collection and processing .....	25
1-3	Incorrectly georeferenced lidar data due to naïve rotational interpolation .....	26
2-1	Optimal orientation of the VLP-16 scanner aboard a UAS .....	39
2-2	Comparison of scan patterns.....	39
2-3	Gaps in the VLP-16 scan pattern .....	40
2-4	Statistical analysis of the VLP-16 scan pattern.....	41
2-5	Point density function relationship to flying height.....	42
2-6	Point density function relationship to forward velocity.....	43
2-7	Plot of gap equation results.....	44
2-8	Plot of gap equation results with missing gap results .....	44
2-9	Plot of overlapping flight lines .....	45
2-10	Effect of rotation rate on scan pattern and gap bands .....	46
2-11	Ground truth of a flight line with nonzero yaw .....	47
2-12	Effects of yaw on point density and gap locations .....	48
2-12	Average scan angle .....	49
2-13	Mission planning for artificially limited maximum range .....	50
3-1	Artifacts in dense matching point clouds.....	67
3-2	Geometry of exposure stations.....	67
3-3	Jonesville Park study site with mission plans .....	68
3-4	University of Florida Agricultural Experimental Station with mission plans .....	68
3-5	Oblique mission simulation .....	69
3-6	Simulated bundle adjustment results .....	70

3-7	Ground sample distance along format as a function of tilt .....	70
3-8	Canopy height models, Millhopper study site .....	71
3-9	Differenced canopy height models, Millhopper study site .....	72
3-10	Histograms of canopy height models.....	72
3-11	Histograms of differenced canopy height models .....	73
3-12	Distortion in dense matching reconstruction of the scene .....	73
4-1	Baseline study site at Ordway Swisher Biological Station near Gainesville, FL .....	94
4-2	Locations and orientations of exposure stations for terrestrial dense matching .....	95
4-3	Oblique view of ACMF dense matching point cloud .....	95
4-4	Longleaf Flatwoods Reserve study site .....	96
4-5	GatorEye Unmanned Flying Laboratory (UFL) .....	96
4-6	Moving window search strategy .....	97
4-7	Circle recognition by Hough transform .....	97
4-8	Flowchart of the stem detection method.....	98
4-9	RANSAC cylinder fitting .....	99
4-10	Steam detection results, TLS data, OSBS study site .....	99
4-11	Outlier ratio trials for RANSAC DBH cylinder fitting for DBH estimation .....	100
4-12	Stem detection results, dense matching cloud, ACMF study plot .....	100
4-13	DBH estimation results, dense matching cloud, ACMF study plot .....	101
4-14	Height estimation results, dense matching cloud, ACMF study plot .....	101
4-15	Stem detection results, lidar cloud, Flatwoods site.....	102
4-16	DBH estimation results, lidar cloud, Flatwoods study plots.....	103
4-17	Height estimation results, lidar cloud, Flatwoods study plots .....	103
A-1	Nominal configuration of VLP-16 in relation to ground.....	116
B-1	Solving for the representative profile .....	123

C-1	Air base .....	126
C-2	Ground sample distance in the direction of tilt.....	126

## LIST OF ABBREVIATIONS

2D	Two-dimensional
3D	Three-dimensional
ACMF	Austin Cary Memorial Forest
AGL	Above ground level
ALS	Airborne laser scanner/scanning
AT	Aerotriangulation
BS/LA	Boresight/leverarm
CCW	Counterclockwise
CDF	Cumulative distribution function
CHM	Canopy height model
COTS	Commercial, off-the-shelf
CW	Clockwise
DEM	Digital elevation model
DBH	Diameter at breast height
DSLR	Digital single-lens reflex
ENU	Easting-Northing-Up
EOP	Exterior orientation parameter
FN	False negative
FP	False positive
FWS	Fixed window size
GCP	Ground control point
GNSS	Global navigational satellite system
GSD	Ground sample distance
GPS	Global Positioning System

IE	Inertial Explorer
IMU	Inertial measurement unit
INS	Inertial navigation system
ITD	Individual tree detection
lidar	Portmanteau of “light” and “radar” (Ring, 1963)
LTP	Local tangential plane
MEMS	Microelectromechanical systems
NED	Northing-Easting-Down
OLS	Ordinary least squares
OSBS	Ordway Swisher Biological Station
PDF	Probability density function
RANSAC	Random sample consensus
RGB	Red-green-blue
RMSE	Root mean square error
RTK	Real-time kinematic
sUAS	Small unmanned aerial systems
SWS	Smoothing window size
SfM	Structure-from-motion
TLS	Terrestrial laser scanner/scanning
TOST	Two one-sided t-test
TP	True positive
UAS	Unmanned aerial systems
UFUAS	University of Florida Unmanned Airborne Systems (Research Program)
VTOL	Vertical takeoff and landing

Abstract of Dissertation Presented to the Graduate School  
of the University of Florida in Partial Fulfillment of the  
Requirements for the Degree of Doctor of Philosophy

**PHOTOGRAMMETRY AND LIDAR FROM THE UNMANNED AERIAL PLATFORM FOR  
MEASURING THE PHYSICAL STRUCTURE OF FORESTS**

By

H. Andrew Lassiter

August 2018

Chair: Benjamin E. Wilkinson

Major: Forest Resources and Conservation

Accompanying the emergence of unmanned aerial systems (UAS) mapping in research and industry has been the development of lightweight sensor formats that, until recently, could only be deployed on larger, manned aircraft. The release of lightweight laser scanners, specifically the Velodyne VLP-16, has brought UAS-borne laser scanning (lidar) to the forefront. The first study simulates and analyzes the VLP-16's peculiar scan pattern at various configurations of flying height, forward speed, and other mission parameters, to ultimately produce guidelines, equations, and software to aid in planning a successful aerial mapping mission with the VLP-16.

Alongside lidar, UAS photogrammetry continues to flourish, particularly in forestry and forest ecology, as UAS platforms and data processing software become easier to use and more affordable. However, the use of UAS photogrammetry has outpaced the development of best practices for mission planning. In most cases, UAS photogrammetry missions resemble conventional aerial photogrammetry missions, despite the photographic and geometric differences between the two cases. Mission planning for UAS photogrammetry stands to benefit from being treated as a case of dynamic, close-range photogrammetry, as opposed to low-altitude aerial photogrammetry. The second study explores the photographic and geometric differences

between conventional aerial and UAS photogrammetry, from which it derives best practices for UAS photogrammetry over forested scenes.

In the final study, an application of UAS-derived point cloud data is presented. UAS mapping via lidar and photogrammetry can provide dense, three-dimensional reconstructions of a forested scene; as the technologies and workflows have developed over the past decade, UAS-augmented forest cruises have begun to rival conventional timber cruises in both efficiency and accuracy. A host of algorithms is presented in the final chapter that automatically detect tree stems in point clouds created from UAS lidar and photogrammetric data, and estimate their locations, diameter at breast height (DBH), and heights. The algorithms' performances are assessed against field survey measurements, with the goal of demonstrating the advantages and shortcomings of estimating pine morphology from point clouds created from the low-altitude aerial pose.

## CHAPTER 1

### INTRODUCTION

At the confluence of Moore's law and the rise of unmanned aerial systems (UAS) lies a unique opportunity for researchers and commercial users alike: access to one's own 3D mapping workflow, from data collection to processing and analysis. Such an endeavor, once undertaken only by those with access to costly aircraft, equipment, and software, has never been more accessible. This democratized approach to mapping opens new doors in forestry, silviculture, and ecology, advancing toward the semi-automated study of the forest at the level of individual trees. What satellites and manned aircraft offer in area of coverage, UAS offer in resolution, allowing users to explore the forest at an ever-finer spatial (and temporal) scale in the 3D point clouds derived from laser scanning and photogrammetry. Virtual forest mensuration from 3D mapping products, from manual measurements to automated model fitting and feature extraction, may soon mature from research to practice.

#### **UAS Lidar**

The first study presented, "Mission planning for low-altitude airborne laser scanning with the Velodyne VLP-16," delves into the unique configuration of the VLP-16 laser scanner. This lightweight sensor, originally developed for self-driving automobiles, has found success in UAS mapping. Velodyne's unique "fan" laser configuration is unlike any laser configuration before it, and so it follows that its scan pattern—more precisely, the pattern of laser pulses incident along some scanned surface—is rather unique as well. The success of the VLP-16 has influenced newer scanners to mimic this configuration, and thus a number of popular sensors exhibit this peculiar pattern of "pointillated affine hyperbolas." This pattern can lead to long strips of poor coverage, thus warranting the study.

One of the reasons for examining the VLP-16’s scan pattern so closely was that parallel studies conducted by the University of Florida Unmanned Airborne Systems Research Program (hereafter referred to as UFUAS) required the use of relatively small targets to be placed in the study areas for the purposes of strip adjustment, boresight/leverarm verification, and absolute accuracy assessment. These custom-made, multimodal targets are corner cube pyramids measuring 1.1 m along the base, which is the diagonal of the three cube faces across which the cube is bisected (Figure 1-1). The planar faces of a pyramid can be used to align overlapping strips of lidar data, and the tip of the pyramid can be used as a ground control point (GCP) for absolute accuracy assessment. In practice, this process compared field survey data gathered via RTK GNSS with coordinates derived from the point cloud data. The top of the pyramid in the point cloud was found by fitting three planes to the three detected planes of the pyramid; the intersection of the three planes was held as the GCP as detected in the point cloud.

## **UAS Lidar Payload Development**

The lidar payload and data processing pipeline was developed by UFUAS. The payload consists of the following components:

1. Velodyne VLP-16 PUCK LITE laser scanner, a lightweight version of the VLP-16 PUCK (590 g vs. 830 g) with VLP-16 Interface Box
2. OEM components of the NovAtel SPAN-IGM-S1 GNSS/INS navigation system (STIM300 MEMS IMU, OEM615 GNSS receiver, MEMS Interface Card)
3. ODROID single-board computer with 1 TB flash storage for logging raw data from both the VLP-16 and the NovAtel SPAN-IGM-S1
4. NovAtel GPS-702-GG Pinwheel GPS antenna (dedicated to NovAtel SPAN-IGM-S1; used for positioning data)
5. Garmin GPS18x antenna (dedicated to the VLP-16 for timestamping lidar data)

The data is then processed according to the flowchart presented in Figure 1-2. Aside from the processing of the navigation data by NovAtel Inertial Explorer (IE) software, all data processing is performed by custom software written by Dr. Ben Wilkinson in the Python language.

### **El-Sheimy Georeferencing Equation**

One function in the software suite is an implementation of el-Sheimy's lidar georeferencing equation (el-Sheimy, 2009) which considers the lidar sensor (S-frame), an integrated GNSS/INS system (b-frame), and the mapping datum (m-frame):

$$\mathbf{r}_i^m = \mathbf{r}_{\text{INS}}^m(t) + \mathbf{R}_b^m(t)(\mathbf{R}_S^b \mathbf{r}_i^S + \mathbf{a}^b) \quad (1-1)$$

The position vector  $\mathbf{r}_i^S$  represents the position of a point  $i$  in the S-frame, i.e. the lidar sensor's Cartesian coordinate system. The vector of interpolated coordinates of the GNSS/INS  $\mathbf{r}_{\text{INS}}^m(t)$  in the m-frame at time  $t$  and the rotation  $\mathbf{R}_b^m(t)$  of the b-frame into the m-frame are both outputs of the processed navigational data. The constant vector  $\mathbf{a}^b$  is the position of the sensor in the b-frame, and the rotation  $\mathbf{R}_S^b$  between the S-frame and the INS b-frame. Together, these are the boresight/leverarm (BS/LA) calibration between the S-frame (laser sensor center) and the center the b-frame, which is physically measured after the construction of the payload.

### **Cubic Spline Interpolation of Navigation Data**

The six exterior orientation parameters provided by the GNSS/INS system at regular time intervals are position  $(X_b, Y_b, Z_b)$  and angular orientation  $(\omega_b, \phi_b, \kappa_b)$ . The lidar sensor and GNSS/INS system collect their data at different rates. The VLP-16 collects up to 600,000 returns per second (dual returns at 300,000 pulses per second), each return with a unique epoch  $t$ . The GNSS/INS collects positional data once per second and inertial data 125 times per second. After processing these data in IE, there exists navigational data at a rate of 125 Hz for epochs  $t_{\text{INS}}$ . Direct georeferencing requires each point in the S-frame to have unique navigational data, thus

requiring an interpolation of the navigational data. By treating each of the six exterior orientation parameters (EOPs) in the navigational data as independent, a one-dimensional cubic spline interpolation can be applied to each parameter (as a function of the lidar epochs  $t$ ).

**Rotational interpolation.** Cubic spline interpolation of rotational data will fail if implemented directly onto the values of the rotational angles  $(\omega_b, \phi_b, \kappa_b)$  because the range of angles is multiply defined; i.e., their ranges are  $[0^\circ, 360^\circ]$  or some translation thereof, and any value outside of this range can be defined also as a value inside the range. If the value of some rotational angle changes such that its value jumps from one extrema of the range to the other between epochs  $t_n$  and  $t_{n+1}$ , all interpolated values between these two epochs will be nonsensical, resulting in failed georeferencing (Figure 1-3). Thus, rotational values must undergo rotational interpolation. Because the sine and cosine functions are continuous functions, they can be exploited for smooth interpolation. Any angle  $\theta$  can be expressed by its sine,  $\sin \theta$ , and cosine,  $\cos \theta$ . The series of sine and cosine values for multiple angles  $\theta$  can then undergo interpolation. The interpolated values for angles  $\theta$  at any interpolated epoch  $t$  can then be found using full circle inverse tangent:

$$\theta_t = \tan^{-1} \left( \frac{\sin_t \theta}{\cos_t \theta} \right) \quad (1-2)$$

A properly-implemented full-circle inverse tangent function will output angles between  $[0^\circ, 360^\circ]$ .

**Example.** Apply a linear interpolation between the angles  $\theta_1 = 359^\circ$  and  $\theta_2 = 2^\circ$  using both naïve interpolation and the rotational method listed above, to find the value of  $\theta_{1.5}$ .

The correct answer can be found intuitively—if at time  $t = 1$  the angle  $\theta_1 = 359^\circ$  is observed, and at time  $t = 2$  the angle  $\theta_2 = 2^\circ$  is observed, the best estimate for the value of  $\theta$  at time  $t = 1.5$  is  $0.5^\circ$ . In this special case, the interpolated epoch is the midpoint between two

epochs. The naïve linear interpolation for this simple case can be solved by taking the mean of the two given values:  $(359^\circ + 2^\circ)/2 = 180.5^\circ$ , which is obviously incorrect. Rotational interpolation, however, will yield the following:

$$\sin \theta_1 = -0.0175; \cos \theta_1 = 0.9998$$

$$\sin \theta_2 = 0.0349; \cos \theta_2 = 0.9994$$

The linear interpolation between  $\sin \theta_1$  and  $\sin \theta_2$  by the naïve method is 0.0087, and between  $\cos \theta_1$  and  $\cos \theta_2$  is 0.9996. Thus:

$$\theta_{\text{interp}} = \tan^{-1} \left( \frac{0.0087}{0.9996} \right) = 0.5^\circ$$

### **UAS Photogrammetry**

The next study, “Oblique UAS photogrammetry in forested scenes,” revisits the practice of conducting low-altitude UAS photogrammetric missions as scaled-down versions of their high-altitude counterparts, where images are collected from a conventional aircraft. With the drastically decreased altitude comes new considerations of the geometric relationship between exposure stations—the location and angular attitude of the camera when an image is taken—and the scene being imaged. This study compares the convention—downward-looking, or nadir, imagery—to tilted, or oblique, imagery (Figure 1-4), a configuration that has been used with great success in urban settings. This study, along with the one presented before, have in common the aim of presenting best practices for planning UAS data collection missions with these two sensors, the miniature laser scanner and the commercial, off-the-shelf digital camera.

The study utilizes both quantitative and qualitative comparisons of the data collected from the two aerial camera poses. For the quantitative study, both nadir and oblique datasets were georeferenced using the same control points, and an accuracy assessment was conducted using twenty checkpoints in the scene. The multimodal corner cube targets (Figure 1-1) served as

the checkpoints. For the qualitative study, canopy height models (CHMs) were generated over a plot of planted pines from lidar, nadir imagery, and oblique imagery. Using the lidar CHM as control, the data from the two aerial poses are assessed.

### **Forest Mensuration from Point Clouds**

The final chapter, “Semi-automated estimation of pine morphology from low-altitude 3D mapping,” is a case study in an application of 3D remote sensing in a forested area—virtual forest mensuration. The study presents a suite of algorithms that operate on the 3D point cloud data derived from these sensors in an effort to demonstrate the capabilities (and limitations) of estimating key forest parameters at the single tree level of precision. The UAS platform’s geometric advantage of low altitude provides lower ground sample distances (GSDs) for images and a much higher point density of lidar data. However, the lightweight sensors required for UAS data collection collect lower-quality data when compared to larger sensors used aboard conventional aircraft. In the balance between the lightweight sensors’ advantages and disadvantages lies the scope of their current abilities to deliver data capable of forest mensuration at the single-tree level.

The latest developments in UAS mapping have remarkable implications for the future of forest inventory and mensuration, but they by no means supplant established methods and technologies. The complexity of the form of trees, from the snarled sand live oak to the straight and simple slash pine, eludes full expression in even the finest-resolution point cloud. This complexity, however, has not deterred this researcher from pondering their form and struggling to understand and express them in prose, tables, graphs, images, and animations, a series of endeavors that has carried nascent ideas into completed studies. May the work presented herein be found to be worthy of the standard of doctoral research at the University of Florida, and that

the algorithms and software developed, results obtained, and questions posed will benefit the communities and industries under the canopy of the School of Forest Resources & Conservation.

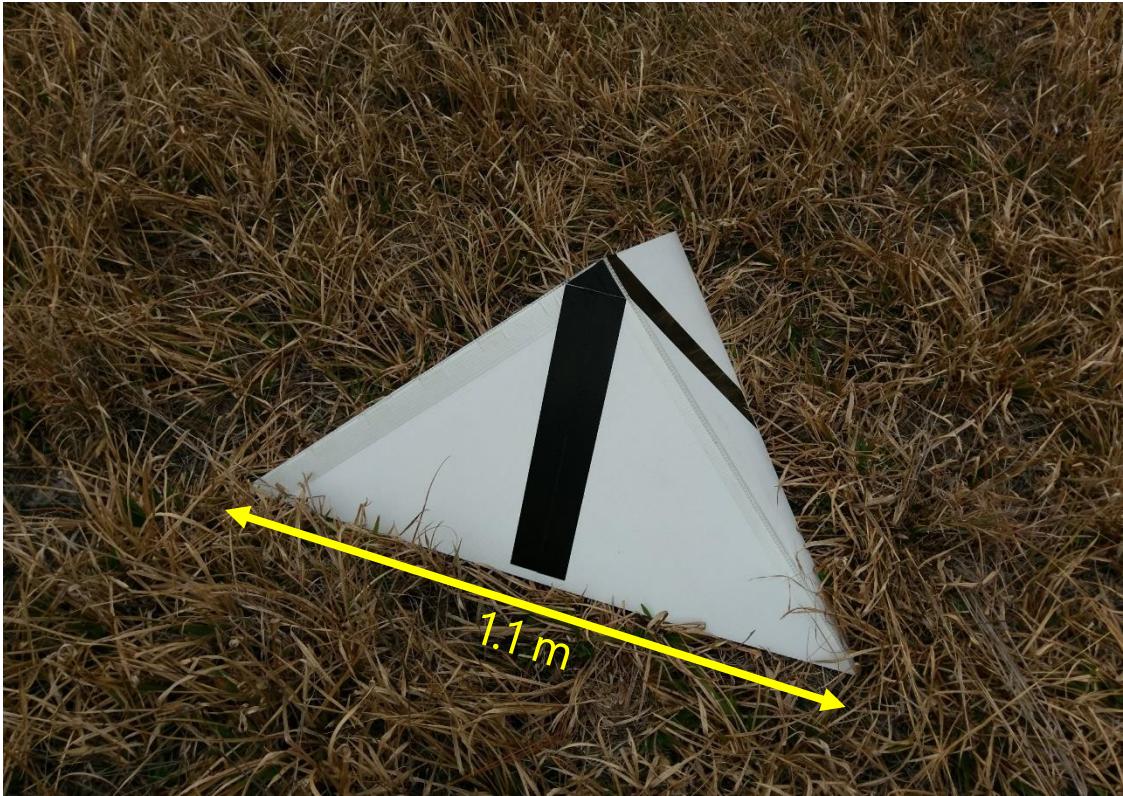


Figure 1-1. Multimodal corner cube target. Photo courtesy of author.

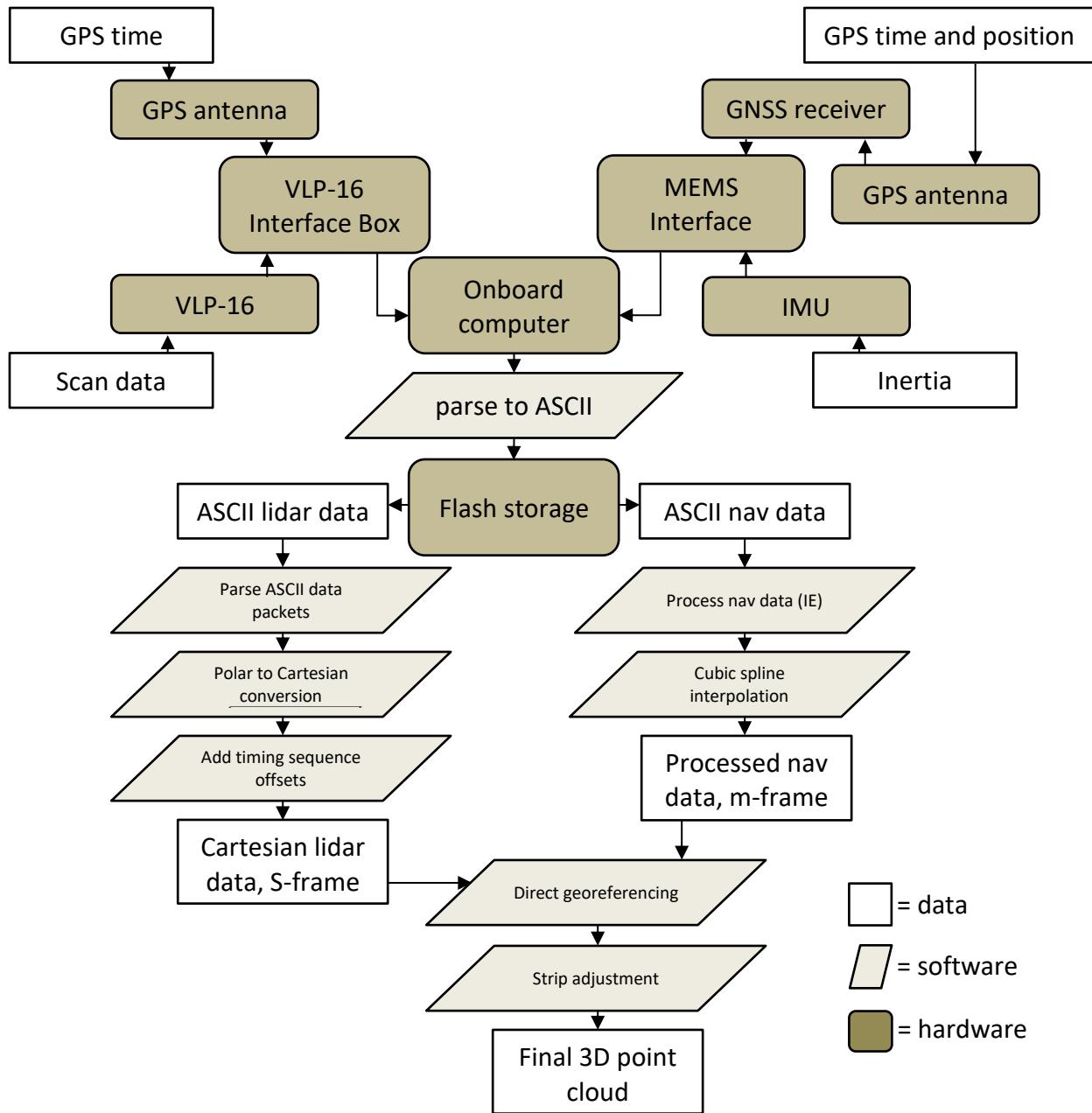


Figure 1-2. Flowchart of lidar data collection and processing.

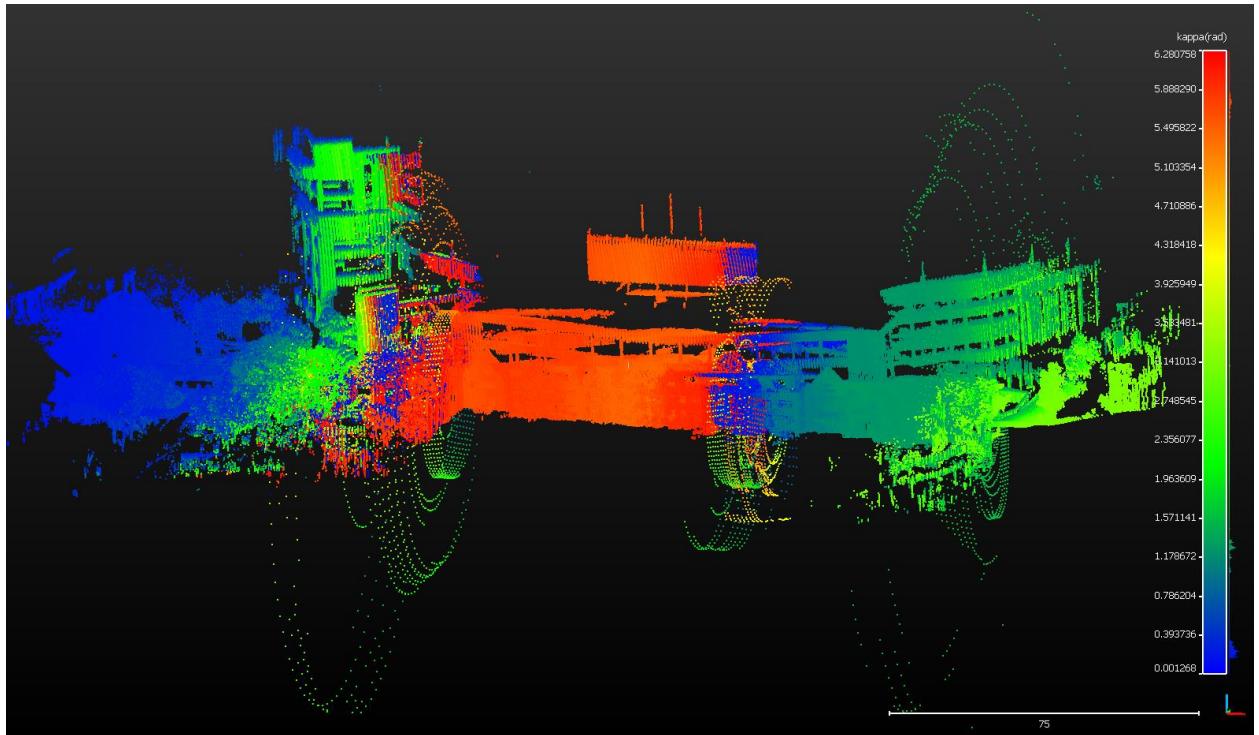


Figure 1-3. Incorrectly georeferenced lidar data due to naïve rotational interpolation. The data are colored by the interpolated value of kappa. As the value of kappa switched between the two extrema of the domain, naïve interpolation failed. Data collected from a mobile terrestrial test near Ben Hill Griffin Stadium, Gainesville, FL.

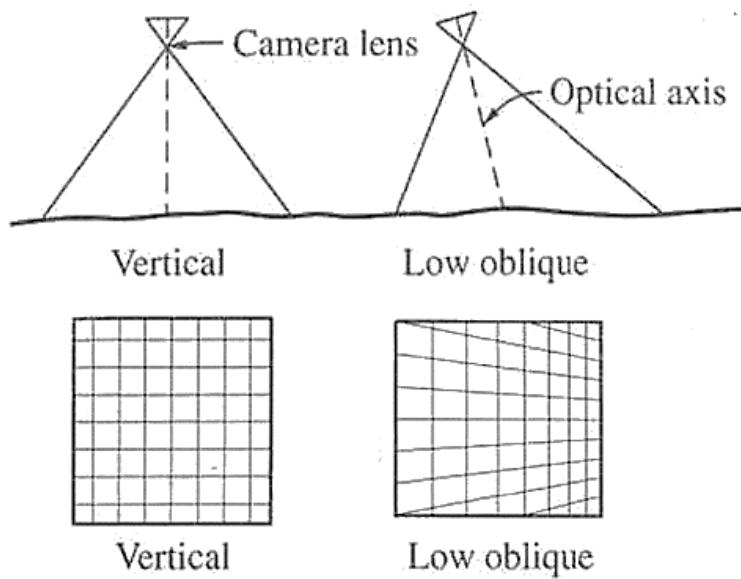


Figure 1-4. Vertical (nadir) and oblique camera poses.

## CHAPTER 2

### MISSION PLANNING FOR LOW-ALTITUDE AIRBORNE LASER SCANNING WITH THE VELODYNE VLP-16

Airborne laser scanning (or lidar) from small, unmanned aerial systems (sUAS) is gaining the reputation as a viable mapping tool for both academic researchers and commercial users within the past decade (Pilarska *et al.*, 2016; Starek and Jung, 2015; Lin *et al.*, 2011). Accompanying the release of the lightweight Velodyne VLP-16 in 2014 has been the emergence of highly-accurate, lightweight navigational sensors (for example, the NovaTel SPAN-IGM-S1 GNSS/INS). These two core components form a payload that is functionally similar to the airborne lidar payloads aboard manned aircraft, but at a fraction of the weight and cost. Commercial outfits (such as YellowScan and Phoenix Aerial Systems) have offered turnkey sUAS lidar systems which utilize the VLP-16 since at least 2015 (Starek and Jung, 2015). Put another way: From an accessible platform—the sUAS—users can now collect an established, familiar data product—the lidar point cloud—that is compatible with existing data processing workflows.

The adaptation of low-altitude airborne lidar, specifically with the VLP-16, presents new issues. The configuration of the scanner head results in a scan pattern quite unlike a conventional line-scanning airborne laser scanner. The VLP-16 emits lasers from sixteen channels oriented in a “fan” of diverging channels, resulting in a 30° vertical field of view. This “fan” of lasers rotates about the scanner’s vertical axis for a 360° horizontal field of view (Velodyne, 2014). This configuration of lasers is functionally similar to the VLP-16’s predecessor, the HDR-64, which was designed for self-driving automobiles (Glennie & Lichten, 2010). To ensure maximum coverage along a flight line when using the VLP-16 in the aerial pose, the most sensible way to orient the scanner is on its side, such that the scanner’s vertical axis is parallel with the direction of travel (Figure 2-1). The resulting scan pattern is a series of hyperbolas, visually distinct from

the scan pattern of conventional line scanners (Figure 2-2). However, the resulting scan pattern does not produce a uniform distribution of laser returns (or points), and in some cases, can produce relatively large gaps in coverage (Figure 2-3). These gaps in coverage are not a function of point density, but rather point clustering, where many laser pulses are striking the target within very close proximity to each other. This stands in contrast to point dispersion, where the laser pulses produce a more evenly-spaced pattern across the target. This concept of point clustering, and what causes this phenomenon in the scan pattern, is investigated below.

Proper mission planning is therefore vital to the success of airborne lidar data collection with the VLP-16. In addition to knowing the capabilities of the aircraft and the direct georeferencing payload, practitioners must also be aware of the scanner's capabilities—maximum range, angular resolution, and beam divergence, to name a few—in order to plan a mission that will yield useable data for the task at hand. A new fold to consider in mission planning with the VLP-16 is its unique scan pattern. This study presents the development of VLP-16 simulation software that allows for the examination scan pattern of the VLP-16 both qualitatively—i.e., manually examining simulated point clouds—and quantitatively, via spatial statistics. Characterization equations for the VLP-16's point density, optimal separation of flight lines, and possible gaps in coverage are also presented.

## **VLP-16 Configuration**

The VLP-16 emits lasers from sixteen channels oriented between  $-15^\circ$  and  $+15^\circ$  in  $2^\circ$  intervals from the scanner's horizontal plane, or xy-plane, resulting in a  $30^\circ$  vertical field of view. The vertical angle  $\omega$  of each channel is fixed and is defined as the counterclockwise angle of the channel with respect to the scanner xy-plane. The channels fully rotate about the scanner's vertical axis, or z-axis, for a  $360^\circ$  horizontal field of view. The lasers are fired one at a time according to a precise timing sequence in which one laser is fired every  $2.304\ \mu\text{s}$ ; after all

sixteen lasers have fired, there is a recharge period of 18.43  $\mu\text{s}$  (Velodyne, 2014). Therefore, each laser firing has a unique time, and because the scanner head is in constant rotation, each laser firing has a unique angle of rotation about the z-axis, or azimuth  $\alpha$ . Note that azimuth is defined as the clockwise angle of rotation about the +z-axis, where the +y-axis is set to zero (Velodyne, 2014). Assuming the configuration described above, over flat terrain, the resulting scan lines from the VLP-16 are affine hyperbolas; as the scanner travels forward parallel to the terrain, these overlapping sets of skewed hyperbolas result in areas of varying point dispersion (Figure 2-3). Laser pulses are emitted at a constant angular interval (or, put more precisely, emitted at a constant time interval while rotating at a constant angular velocity) along the scanner's horizontal field of view. Under the presented configuration with the VLP-16 turned on its side, point density along a nominally flat surface is a function of linear distance across the scanning profile (Figure 2-4). The point density, along with the point clustering, are the primary considerations for planning a UAS data collection mission with the VLP-16.

### **Analytical Characterization of VLP-16 Scan Pattern**

#### **Point Density Function**

The point density of the laser return pattern of the VLP-16, as a function of lateral distance from the flight line  $x$ , can be closely approximated by the point density function

$$p(x) = \frac{l_f h}{2\pi v_z(h^2+x^2)} \quad (2-1)$$

where  $h$  is the height above ground,  $l_f$  is the pulse frequency of the scanner (approximately 300,000 pulses per second), and  $v_z$  is the forward velocity along the scanner's z-axis, assuming the z-axis is parallel to the target plane (see Appendix A). This equation is simplified under the assumption that the laser pulses are emitted at a uniform rate, although in practice, this is not the case. Each of the sixteen channels emits a pulse once per 2.304  $\mu\text{s}$ , followed by a recharge period

of  $18.43 \mu\text{s}$ . This simplification, however, has a negligible effect on the results, as demonstrated by comparing the equation's results to the scan pattern simulation presented in the next section.

**Example.**  $h = 45 \text{ m}$ ,  $v_z = 9 \text{ m/s}$ ,  $l_f = 3 \times 10^5 \text{ pulses/s}$ .

$$p(x) = \frac{(3 \times 10^5 \text{ pulses/s})(45 \text{ m})}{2\pi(9 \text{ m/s})((45 \text{ m})^2 + x^2)}$$

The plot of this example function is found in Figure 2-5. The maximum point density nadir to the scanner can be found at  $x = 0 \text{ m}$ :

$$p_{max}(x) = p(0) = \frac{(3 \times 10^5 \text{ pulses/s})(45 \text{ m})}{2\pi(9 \text{ m/s})((45 \text{ m})^2)} = 118 \text{ points/m}^2$$

### Optimal Separation of Flight Lines

To assure a minimum density of laser returns along the profile of the laser return pattern of a mission with parallel flight lines, the maximum flight line separation  $w$  can be expressed as

$$w = 2 \sqrt{\frac{l_f h}{\pi p_d v_z} - h^2} \quad (2-2)$$

where the minimum desired return density is  $p_d$ , expressed as points/m<sup>2</sup>. This equation is a further derivation of the point density function, as shown in Appendix A. Inspection of the plot of  $p(x)$  should be used to inform a sensible choice for a value of  $p_d$ .

**Example.**  $h = 45 \text{ m}$ ,  $v_z = 9 \text{ m/s}$ ,  $l_f = 3 \times 10^5 \text{ pulses/s}$ ,  $p_d = 180 \text{ points/m}^2$ .

$$w = 2 \sqrt{\frac{(3 \times 10^5 \text{ pulses/s})(45 \text{ m})}{\pi(180 \text{ points/m}^2)(9 \text{ m/s})} - (45 \text{ m})^2} = 50 \text{ m}$$

### Gap Equation

The bands of gaps present in the laser return pattern occur at certain lateral distances from the flight line  $x_i$ , and can possibly occur at the distances found using the equation

$$x_i = h \tan \left( \cos^{-1} \left( \frac{rh \tan \Delta\omega}{iv_z} \right) \right) \quad (2-3)$$

where  $r$  is the rotation rate of the scanner head (typically 5-20 Hz),  $\Delta\omega$  is the angular separation between adjacent channels ( $2^\circ$  for the VLP-16); and  $i$  is an integer, indicating the  $i^{th}$  gap outward from the flight line at  $x = 0$ . For smaller values of  $i$ , the term for which the arccosine is taken can be  $\geq 1$ , which yields a complex value of  $y_i$ . Example results from the gap equation for various flying height and forward speeds can be found in Tables 2-1:3.

### Simulation of the VLP-16 Scan Pattern

To test the VLP-16 scan pattern in a controlled environment, software was created in the MATLAB programming language to simulate the laser return pattern of the scanner in the aerial pose. The scanner is modeled as a point at a user-specified height above a horizontal plane, traveling a constant speed (also specified by the user) along a vector parallel to the plane. The emitted lasers are modeled as lines passing through the scanner point toward the target plane. The “vertical” angles of the lasers (referred to as “channels” by the manufacturer) are fixed, as described above. The “horizontal” angles of the lasers, or azimuths, are determined by the orientation of the scanner head at each epoch as it rotates about the scanner’s vertical axis. For each set of sixteen laser pulses emitted at epoch  $t$ , the simulation passes the line parameters to a solver, which finds the intersections of the lines with the target plane. The coordinates of those intersections—a simulated point cloud—are recorded and saved in a text file. Further explanation of the simulation software can be found in Appendix B.

### Input and Output

The software takes as input key mission parameters with regards to the orientation and operation of the scanner: height above ground, forward speed, scanner head rotation rate, and yaw (i.e., the difference in the scanner’s +z-axis and the direction of travel). The output is a point cloud that is a sample of the laser pulse return that could be expected from a mission flown over

flat ground under these mission parameters. The point cloud output is a comma-delimited text file, where each row represents one return, or point. For each point, the following attributes are attached: the mapping frame coordinates,  $(X, Y, Z)$ ; the azimuth of the scanner head when the pulse was fired,  $\alpha$ ; the vertical angle, or channel, from which the laser was fired,  $\omega$ ; the time at which the laser was fired,  $t$ ; the range of the laser return (i.e., how far the laser traveled before intersecting the ground),  $\rho$ ; and the direction in the mapping frame that the laser was fired,  $\langle dir_X, dir_Y, dir_Z \rangle$ . (Definitions of azimuth and vertical angle can be found in the previous section.)

Visual inspection of the simulated point clouds generated at various heights and speeds reveal a mesmerizing pattern (Figure 2-3). Most notable about this pattern are the bands of point clustering that run parallel to the direction of the scanner's travel. These bands are not the result of a lower point density, but rather point clustering. The scan pattern analysis and gap equation seek to detect and predict the location of these bands as a function of the mission parameters of flying height, flying speed, and the rotation rate of the scanner head (which can be adjusted anywhere from 5 Hz to 20 Hz). Predicting where these bands of high clustering occur as a function of mission parameters will lead to recommendations for optimal values for the parameters mentioned above, as well as side lap (i.e., the overlap of adjacent strips).

### **Spatial and Error Analysis of Simulated Data**

The scan pattern, for all of its peculiarities, is in fact a repeating pattern. This allows for the extraction of a narrow across-track profile which can be used for statistical analysis. The apparent clustering and dispersion of the points along the profile is analyzed by binning the profile of the simulated point cloud and calculating the nearest neighbor index for each bin. The nearest neighbor index is a measure applied to point patterns that indicates whether the pattern is

clustered, random, or dispersed. First, for each simulated return (point) in the bin  $s_i$ , its nearest neighbor and the distance between the two ( $d_{min}$ ) is found. This is accomplished by calculating a distance matrix (of dimensions  $n \times n$ ) for  $n$  points in the bin, and for each column in the matrix, the lowest off-diagonal value is retained. The mean of those  $n$  nearest neighbor distances is the observed nearest neighbor distance, or  $\bar{d}_{obs}$ :

$$\bar{d}_{obs} = \frac{\sum_{i=1}^n d_{min}(s_i)}{n} \quad (2-4)$$

Next, the expected nearest neighbor distance  $\bar{d}_{exp}$  is calculated, which is a function of the area  $A$  of the bin:

$$\bar{d}_{exp} = \frac{0.5}{\sqrt{n/A}} \quad (2-5)$$

Next, the z-score for each bin is found:

$$z = \frac{d_{obs} - d_{exp}}{SE} \quad (2-6)$$

where  $SE = 0.26136/\sqrt{n^2/A}$ .

The z-score provides a normalized measure of the point dispersion or clustering in the bin. For example, a z-score of  $\leq -1.96$  (with its corresponding p-value of  $\leq 0.05$ ), indicates with 95% or greater likelihood that the point pattern is clustered, while a z-score of  $\geq +1.96$  ( $p \geq 0.95$ ) indicates a 95% or greater likelihood that the pattern is dispersed, i.e. approaching even spacing amongst points (Clark and Evans, 1954; O’Sullivan and Unwin, 2010). (This interpretation of the z-score presented is reflective of the source material, but is not appropriate for analyzing data of this nature; see Discussion.)

The analysis of the scan pattern extends beyond its geometric pattern. The point cloud simulation software records other attributes of each simulated return, such as the position of the scanner, range from scanner, time, and direction of the simulated laser pulse. These attributes can

provide a simulation of not only of point density and dispersion, but also the expected absolute accuracy of the points. The information that can be gleaned from these attributes—e.g., range and incidence angle—are key components of the lidar error budget (Baltsavias, 1999). Error propagation within the scanner’s frame (i.e., error related to the scanner itself, relative to the scanner’s coordinate system, not accounting for error from the navigational sensor data) is essential to understanding the reliability of the resulting point cloud. For this study, the simulated returns within each bin are analyzed according to attributes unique to each return. Within each bin, histograms can be generated to show the distribution of the number of returns as a function of average range and average incidence angle for each bin. These two measures are directly proportional over the ideal target plane used in this study’s simulation; as the incidence angle of a laser pulse “increases” (i.e. departs from being normal to the target), so too increases the range that pulse must travel to reach the target. An analog to the accuracy of the points within a given scanned area can be visualized by finding the distribution of ranges and scan angles across the scan profile.

## **Results and Discussion**

### **Single Strip**

As shown in the point density function equation, the point density of the scan pattern is inversely proportional both to the flying height and forward velocity of the scanner. These relationships are shown in Figure 2-5 and 2-6, respectively. The simulations are in agreeance with the point density equation and, in most cases, the gap equation. There are, however, three limitations of the gap equation made evident through simulation:

1. Any gaps present at or near nadir to the flight line (i.e., gaps at some low value of  $x_i$ ) are not always reported by the gap equation.
2. The severity of the gaps at some lateral distance  $x_i$  is not reflected in the results of the gap equation. The equation only reports the possibility of gaps occurring at some lateral distance.

3. At higher forward velocities, some significant bands of gaps may not be reported by the gap equation.

An example of these limitations is shown in Figures 2-7 and 2-8. In conjunction with the gap equation results, inspecting the point density histograms for each bin's z-score (Equation 2-6) reveals flaws in interpreting the z-score as a means of detecting gaps. This is discussed more in the Spatial and Error Analysis section below.

As observed through simulation, in the areas of the greatest degree of point clustering, the gaps in the scan pattern tend to present as either near-rhombuses, narrow and elongated perpendicular to the direction of flight. As the lateral distance from the flight line  $x_i$  increases, the gaps will sometimes present as somewhat linear clusters of points, which have a less deleterious effect on the coverage of the area. Gaps near nadir to the flight line are elongated to the point of presenting similarly to a widely-spaced linear scan pattern, especially as the flying height increases (and the eccentricity of the hyperbolas decrease).

## Overlapping Strips

With the exception of one-way mission plans along some corridor, such as a utility easement or transportation corridor, most lidar data collection missions using UAS will likely feature parallel, overlapping strips of data. These overlapping strips not only provide common targets in each strip which can be used for strip adjustment and accuracy assessment, but also can be used to “fill in” the gaps present in the VLP-16 scan pattern. Using the optimal flight line separation equation, it is possible to plan a mission in which a desired minimum point density is achieved across the mission area most efficiently. Figure 2-9 shows the resulting point density and point dispersion of a mission flown at 45 m flying height, 9 m/s forward velocity, with parallel flight lines spaced at 50, 68, and 88 m. These flight line spacings are the result of the flight line separation equation for values of  $p_d = 180, 150$ , and  $120$  points/m<sup>2</sup> respectively.

## **Rotation Rate and Gaps**

The results in Tables 2-1:3 are initially misleading. As the rotation rate increases, the number of predicted possible locations of gaps increases proportionally. In fact, as the rotation rate doubles, so too does the number of predicted gaps. The predicted gap locations are somewhat linear, occurring every  $x$  meters along the profile at some rotation rate  $r$ ; at rotation rate  $2r$ , the distance between gaps becomes roughly  $x/2$ . But the severity of those gaps is not reflected in the equation. In fact, as the rotation rate increases, the severity of the gaps decreases notably, and shown in Figure 2-10. This study does not comment on the potential benefits of lowering the rotation rate of the scanner head, but the results do show that a lower rotation rate, especially when coupled with a higher forward velocity, can have detrimental effects on the quality of the resulting coverage.

## **Yaw**

Adding yaw, or crabbing, the scanner with respect to the direction of flight leads to the coverage gaps becoming narrower along track and elongated across track, with respect to the gaps that would occur under similar conditions without crabbing. This is shown in Figure 2-11. The primary deleterious effect of the coverage gaps—increasing the odds of missing small areas of interest, e.g. linear features such as sidewalks or power lines, or small areas of interest such as targets—is negated with the addition of only a small amount of yaw. This has a minimal effect on the width of a single strip, as its width is a function of the cosine of the yaw angle. For example, even a yaw angle of  $30^\circ$  would only yield a  $1 - \cos 30^\circ = 13\%$  reduction in the width of that strip.

The analytical equations presented above can be further generalized to include yaw angle, as demonstrated in the derivations in the final section of Appendix A. Thus, intentionally

crabbing the scanner can be worked into the mission planning. The point density function (Equation 2-1) generalizes to

$$p(x) = \frac{l_f h \cos \alpha}{2\pi v_z(h^2 \cos^2 \alpha + x^2)} \quad (2-7)$$

The flight line separation equation (Equation 2-2) generalizes to

$$w = 2 \sqrt{\frac{l_f h \cos \alpha}{\pi p_d v_z} - h^2 \cos^2 \alpha} \quad (2-8)$$

The gap equation (Equation 2-3) generalizes to

$$x_i = h \tan \left( \cos^{-1} \left( \frac{hr \tan \Delta \omega}{iv_z} \right) \right) \cos \alpha \quad (2-9)$$

The effect of adding yaw to the scanner, and the results of the generalized analytical characterization equations are shown in Figure 2-12.

### Spatial and Error Analysis

The question remains of which (if any) z-score threshold is indicative of point clustering resulting in undesirable gaps in coverage. The z-score interpretation presented above is based on the distribution of expected nearest neighbor distances in a theoretically random distribution; the z-score then indicates the probability that a distribution of points in a given area is clustered, random, or dispersed. The points in the VLP-16's scan pattern are not random events, however. The *p*-values associated with nearest neighbor z-scores are not useful information; thus, the probabilistic interpretation of the z-score is not necessarily applicable. In fact, the z-scores appear to be fully relative, only useful for comparison within a given flight configuration. Inspection of Figure 2-10 shows that both the 20 Hz and 5 Hz flight have comparable peak minimum z-scores of -18 and -21, respectively. This belies the severity of the gap problem in the 5 Hz flight (or, conversely, overstates the gap issue in the 20 Hz flight, which is all but negligible).

One final consideration for mission planning is optimizing the scan angles. Figure 2-13 shows the average absolute values of scan angles of each return within 1-m bins across the profile. Wider spacing of flight lines may lead to fewer flight lines needed to cover an area, but as the scan angle (and range) of the returns increases, so too will their error. Another thought is to simply exclude returns beyond a certain range for the sake of a higher accuracy across the resultant point cloud. Figure 2-14 shows an example scan profile where the maximum range has been limited to 60 m. The profile width decreases drastically, but depending on the desired accuracy of the data, this may be advantageous. Note that the optimal flight line separation equation cannot be used if the maximum range is limited in this manner.

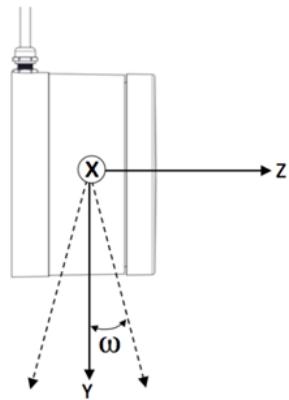


Figure 2-1. Optimal orientation of the VLP-16 scanner aboard a UAS. This figure is a modified version of one found in the VLP-16 User Manual (Velodyne, 2014).

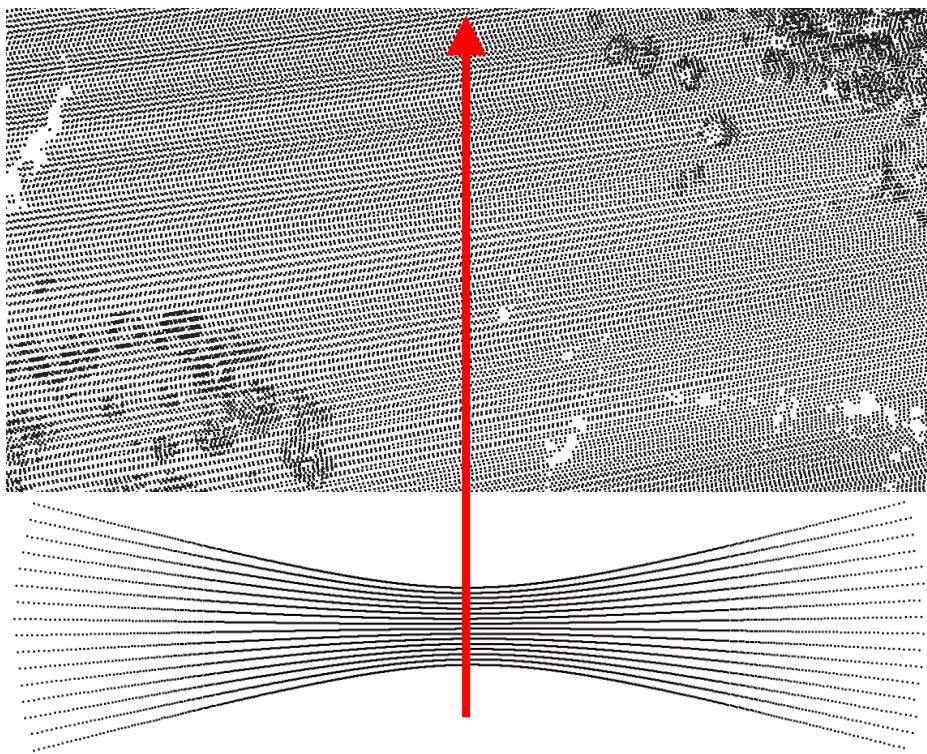


Figure 2-2. Comparison of scan patterns. The linear scan pattern of a conventional ALS line scanner (top) is apparent over nearly flat terrain. The scan pattern of a single pass of the VLP-16's sixteen lasers over a target plane at a flying height of 30 m (below) reveals a much different pattern. Each laser traces a path of a hyperbola across the plane, and the forward motion of the scanner adds a slight affine distortion to each hyperbola. Direction of flight shown in red.

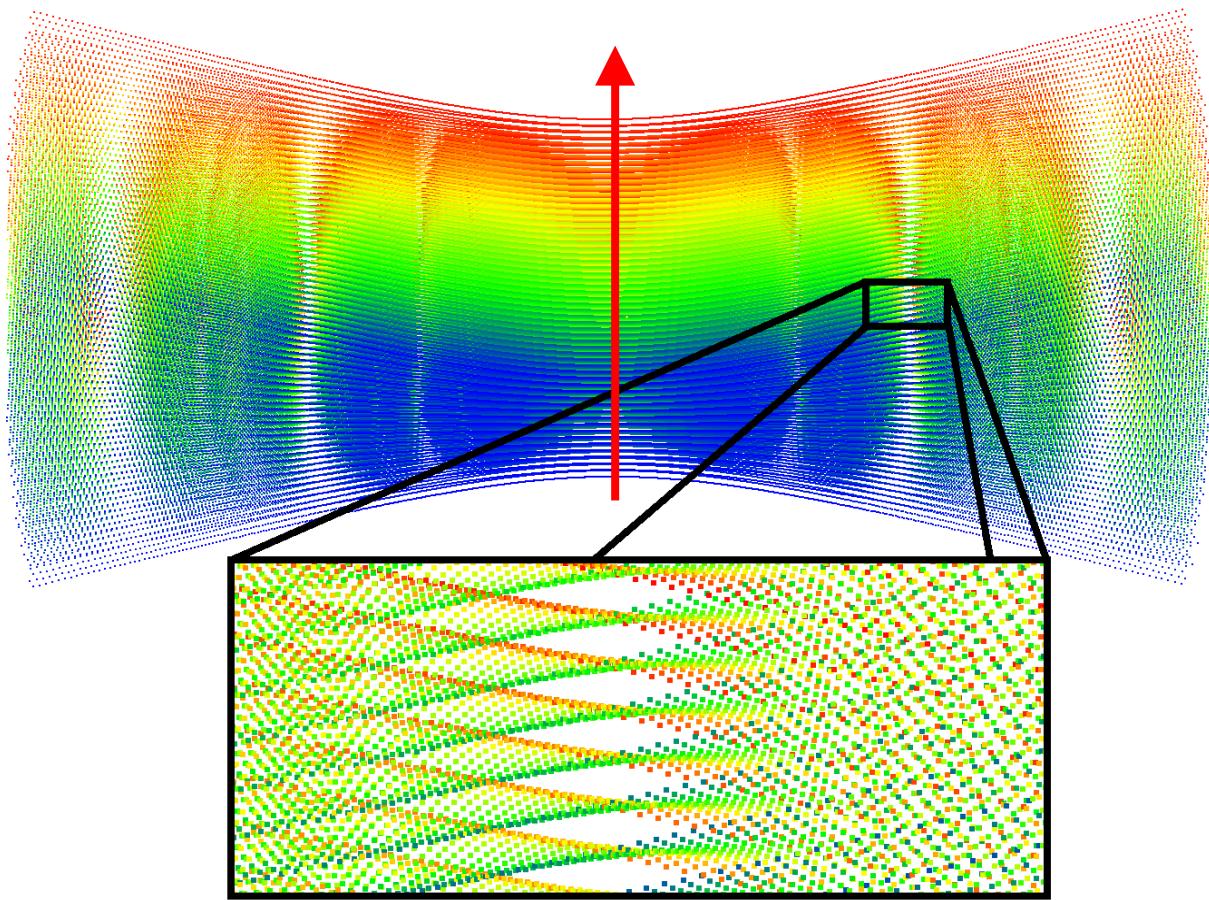


Figure 2-3. Gaps in the VLP-16 scan pattern. A simulated point cloud of the VLP-16 flying 30 m above a target plane at a speed of 10 m/s. The detail shows an example of the gaps in the point coverage that result from the unique scan pattern. The above diamond-shaped gaps are roughly  $1 \text{ m} \times 2 \text{ m}$  each. The points are colored by time of (simulated) return (blue > green > yellow > red).

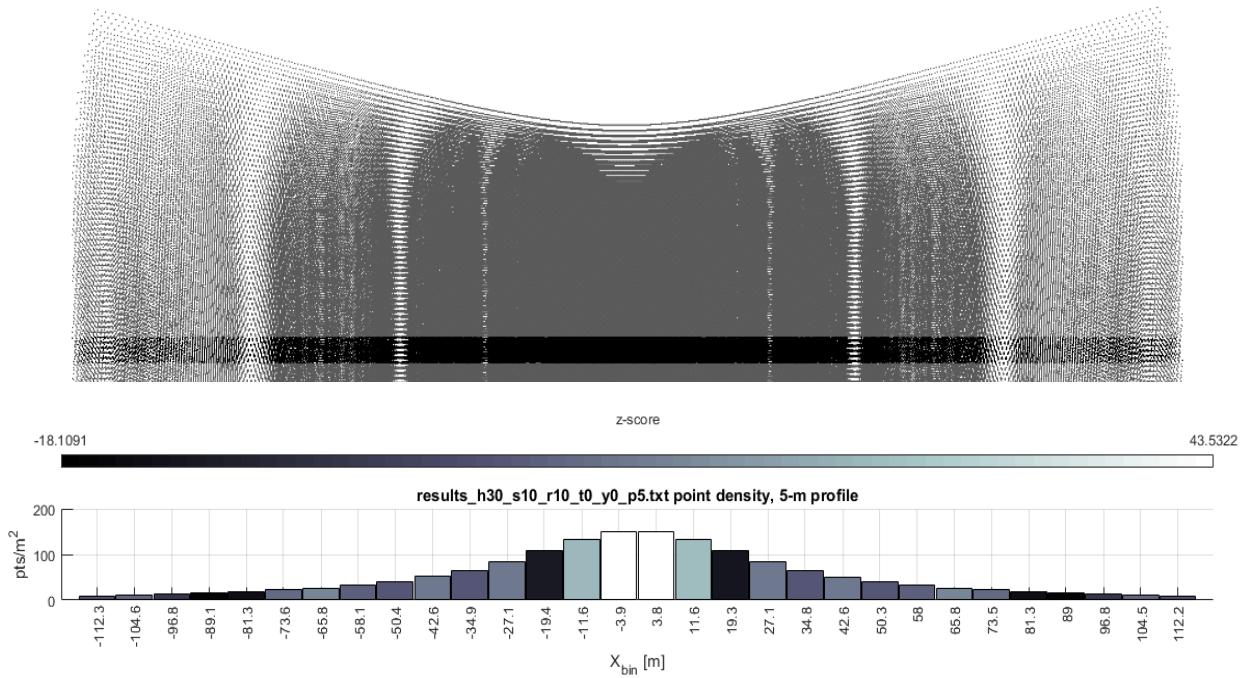


Figure 2-4. Statistical analysis of the VLP-16 scan pattern. Because the VLP-16 scan pattern is repeating, a representative cross-track profile of the scan pattern can be extracted to use for spatial statistical analysis. The black stripe above depicts a 5-m-wide profile in the direction of flight. The histogram depicts the location and width of the bins. Each bin is colored by its clustering z-score.

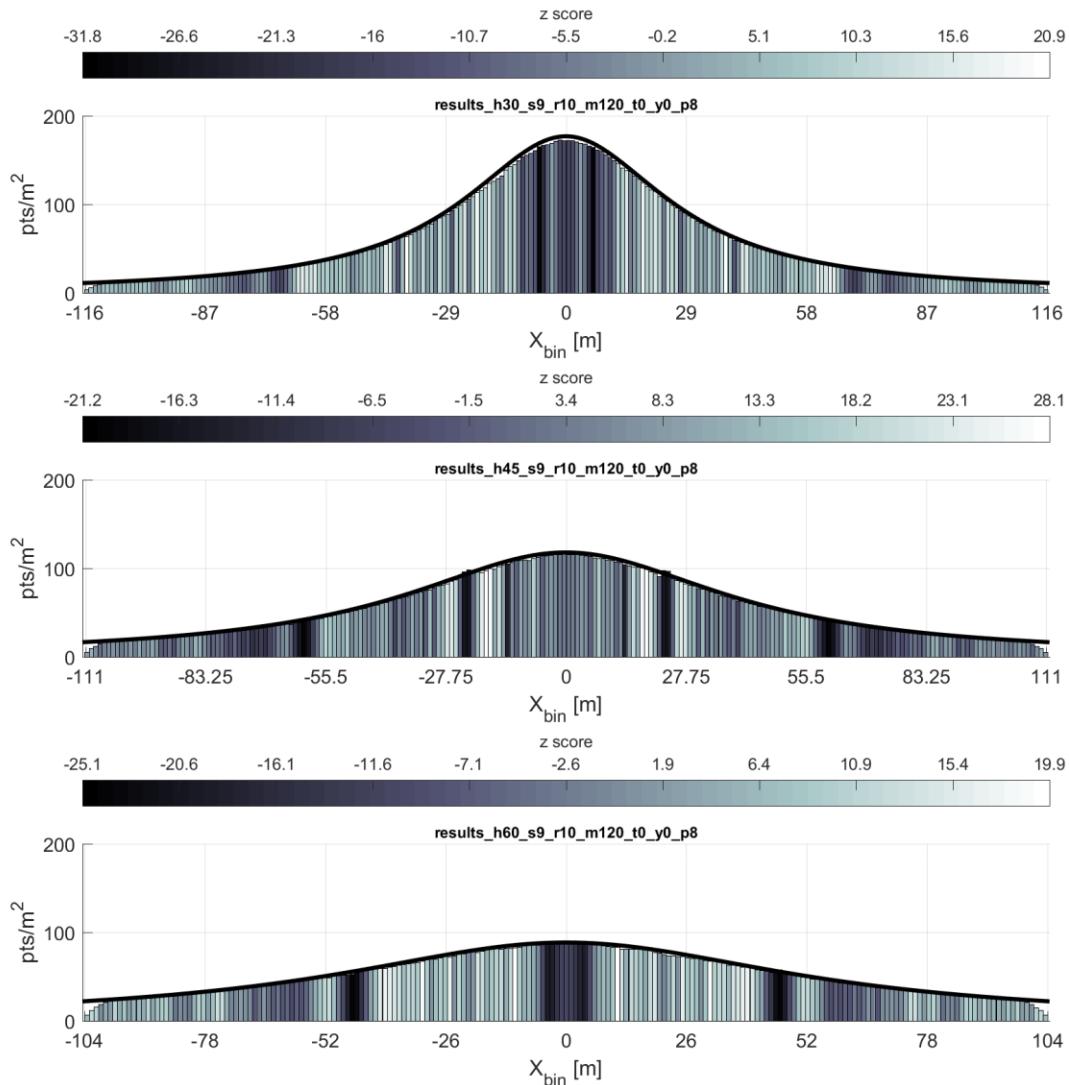


Figure 2-5. Point density function relationship to flying height. From top to bottom, point density function (black line) and simulation results (histogram) for flying heights of 30 m, 45 m, and 60 m. Forward velocity held at 9 m/s

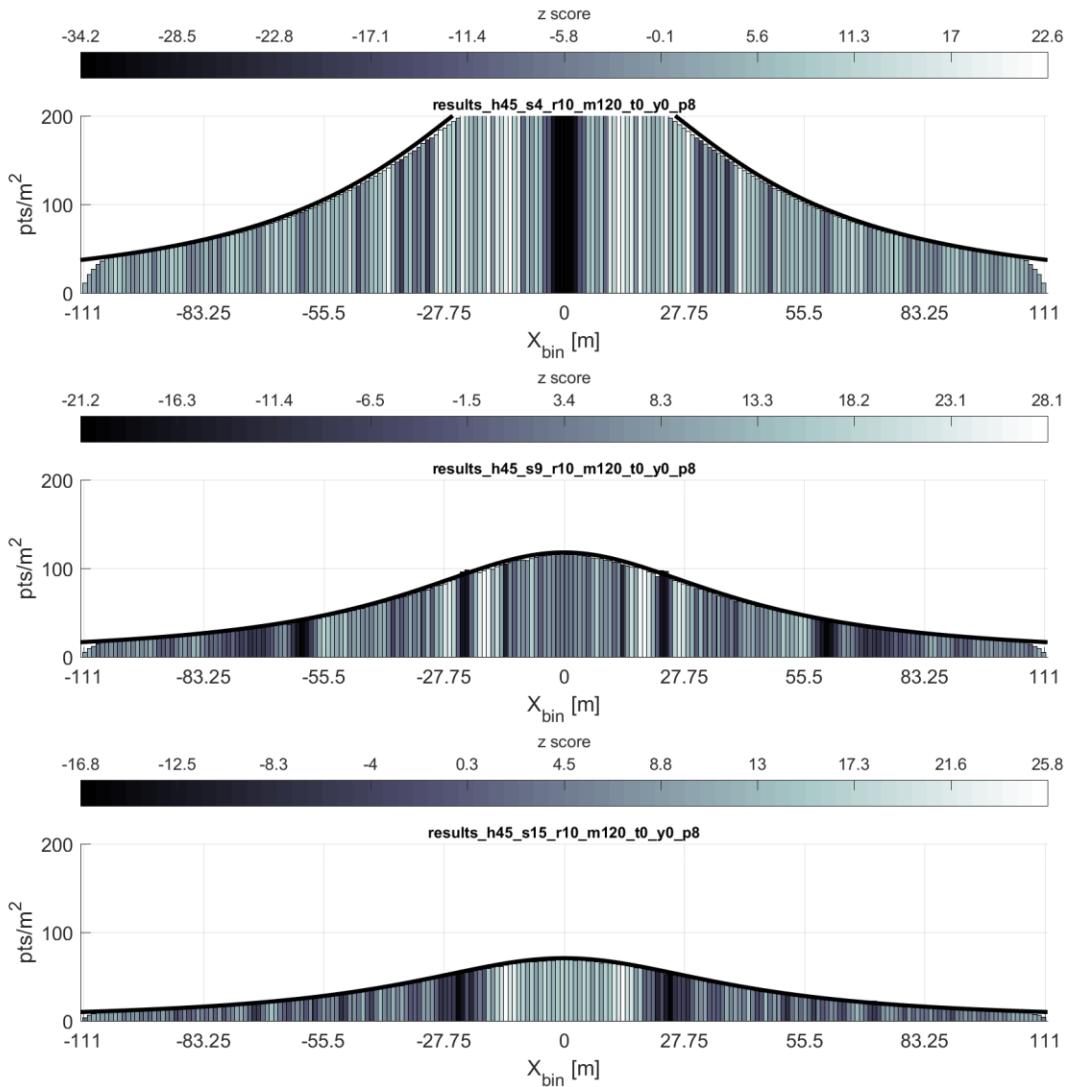


Figure 2-6. Point density function relationship to forward velocity. From top to bottom, point density function (black line) and simulation results (histogram) for forward speeds of 4, 9, and 15 m/s. Flying height is held at 45 m. The y-axis is truncated for the 4 m/s flight for sake of comparison to the other profiles.

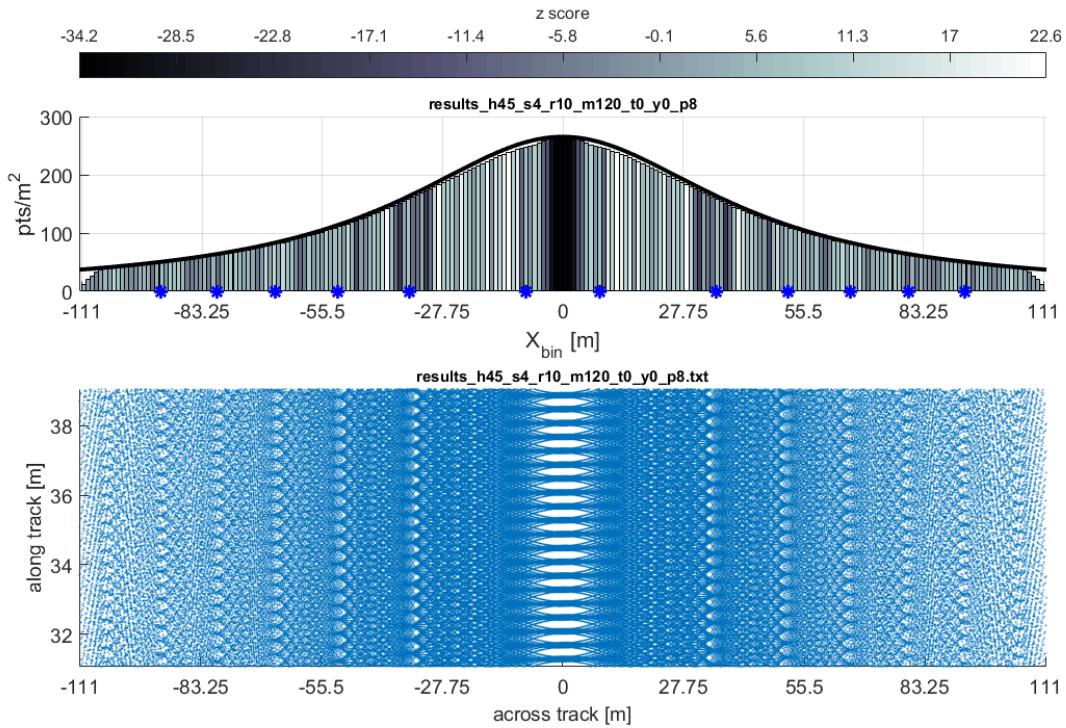


Figure 2-7. Plot of gap equation results. Flying height 45 m, forward velocity 4 m/s. TOP: Gap locations reported by the equation are marked with blue asterisks. BOTTOM: Return pattern of the same flight, exaggerated in the y-direction to show detail.

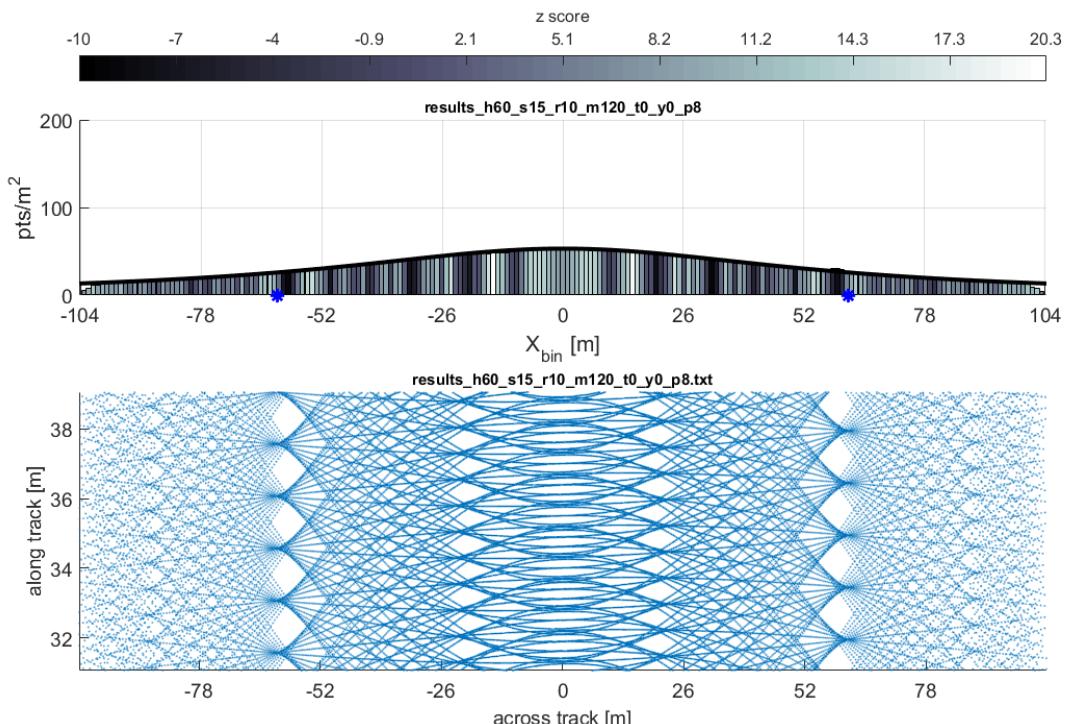


Figure 2-8. Plot of gap equation results with missing gap results. Flying height 60 m, forward velocity 15 m/s.

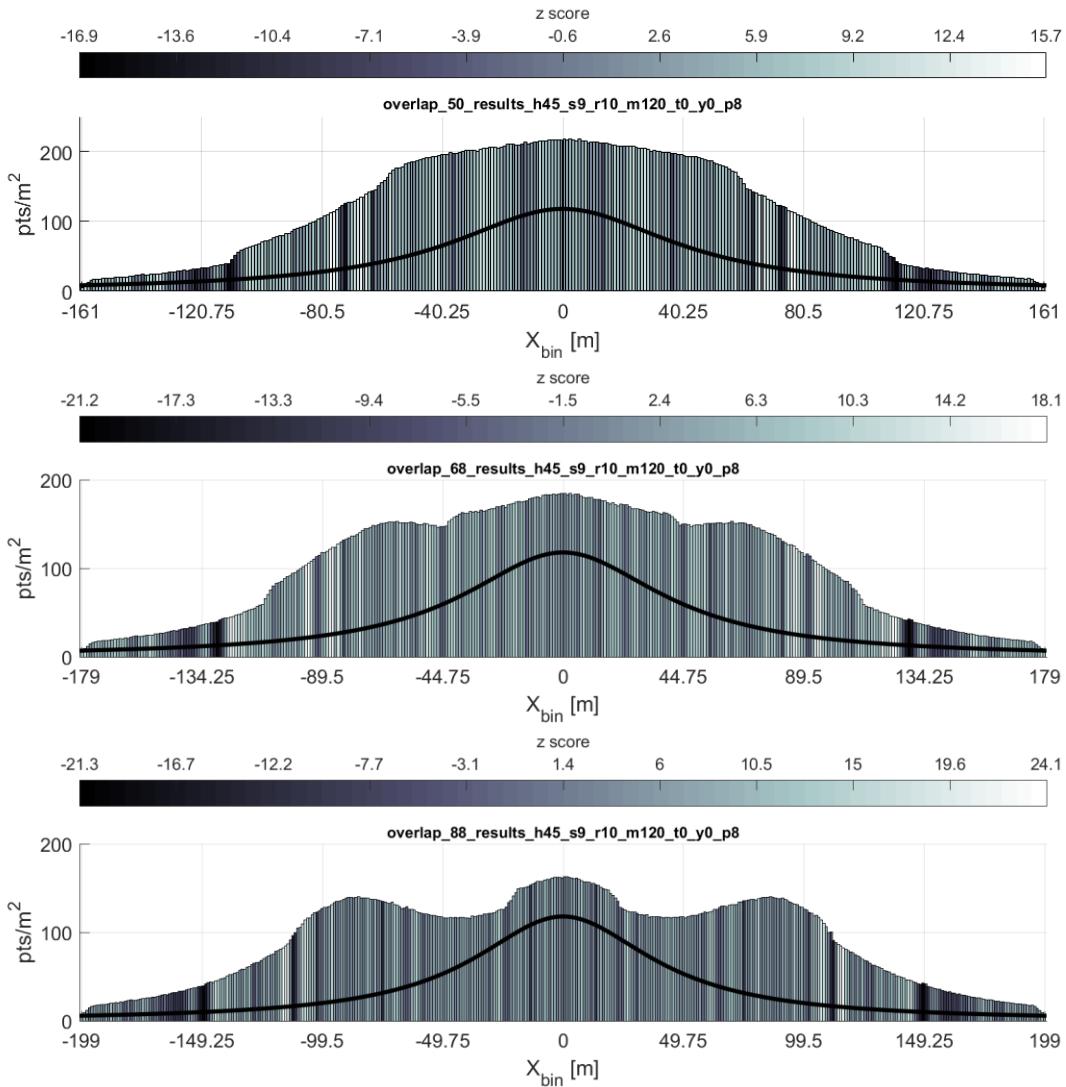


Figure 2-9. Plot of overlapping flight lines. Flying height 45 m, 9 m/s forward velocity, with parallel flight lines spaced at (top to bottom) 50, 68, and 88 m. These flight line spacings are the result of the flight line separation equation for values of  $p_d = 180$ , 150, and 120 points/m<sup>2</sup> respectively. Point density function of a single strip shown (black line).

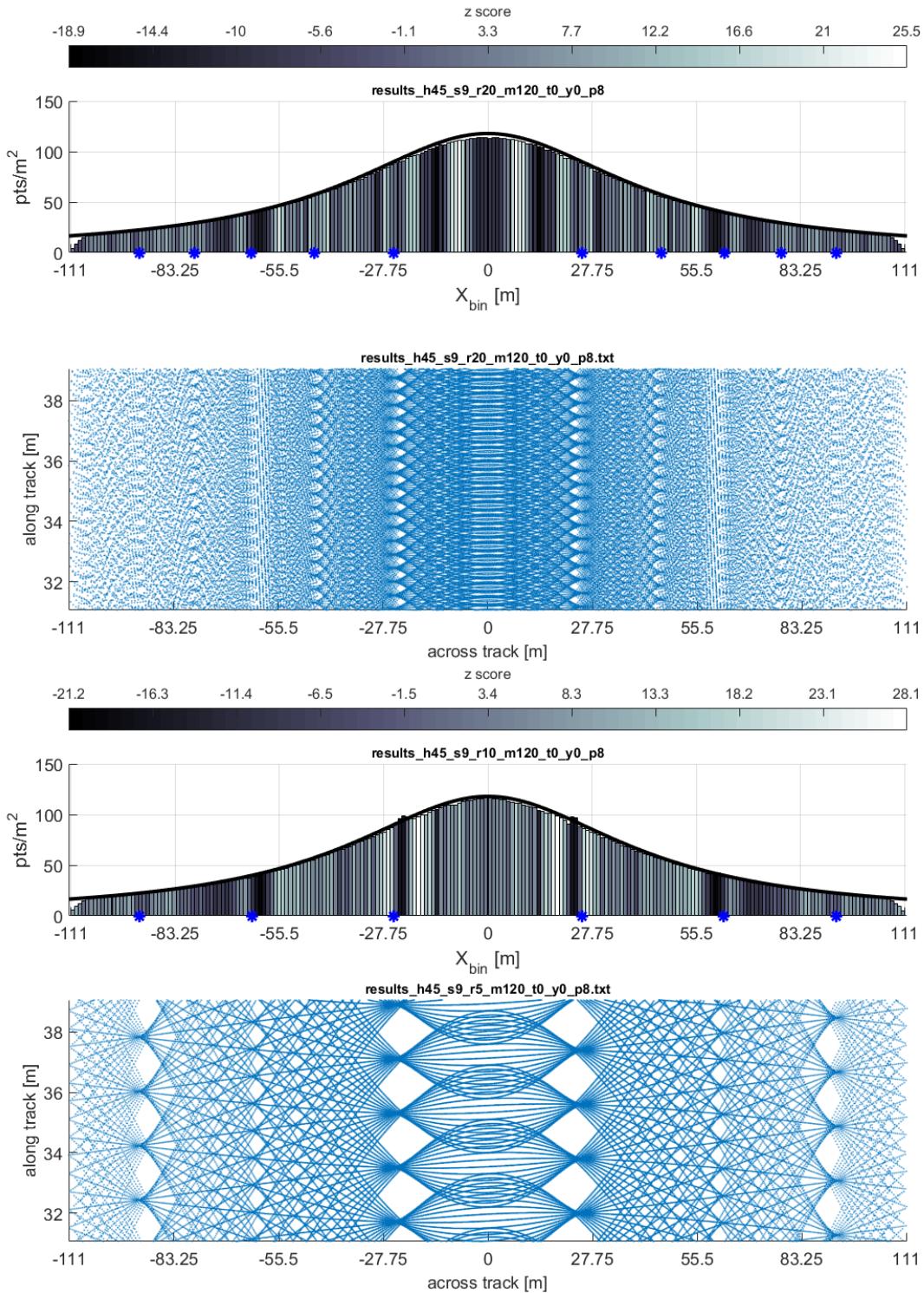


Figure 2-10. Effect of rotation rate on scan pattern and gap bands. Though the number of predicted gaps decreases as the rotation rate of the scanner head decreases, the severity of the gaps becomes notably greater. TOP: Flying height 45 m, forward velocity 9 m/s, rotation rate 20 Hz. BOTTOM: Rotation rate 5 Hz.

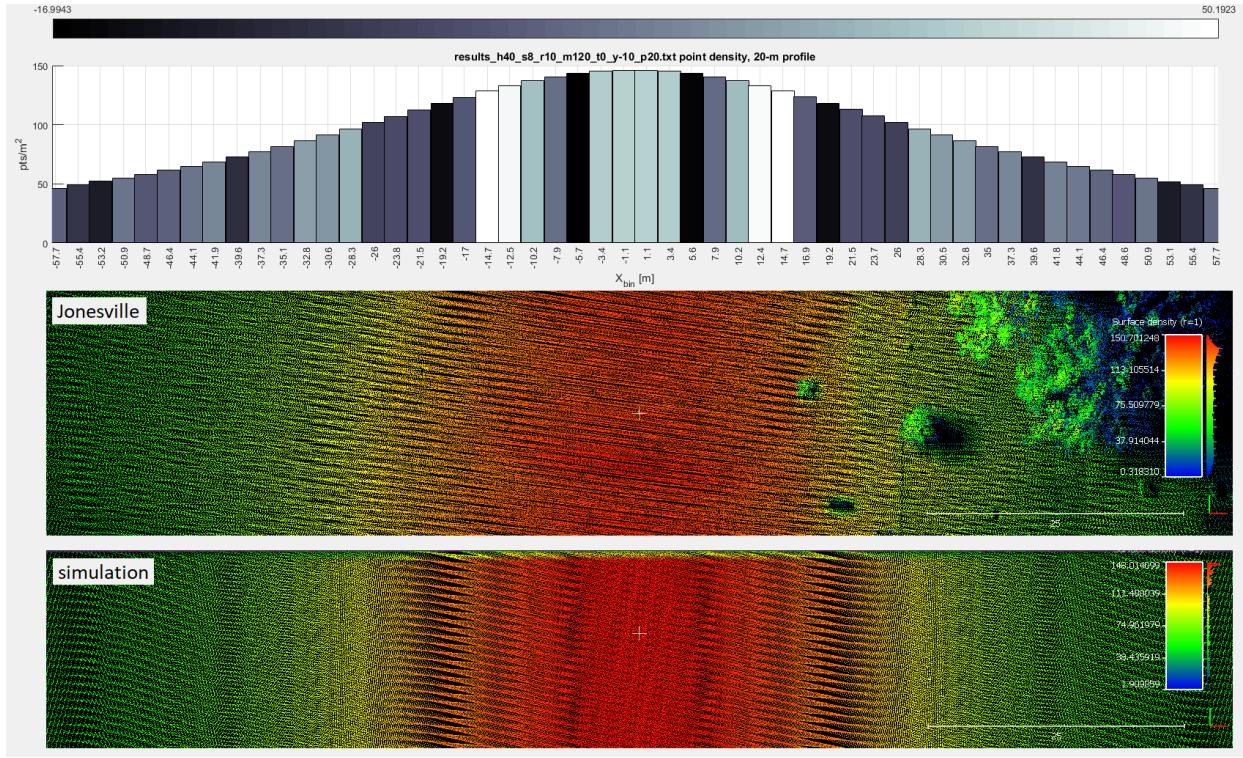


Figure 2-11. Ground truth of a flight line with nonzero yaw. This figure depicts the results of a test flight (middle) that was flown with approximately  $-10^\circ$  of yaw, versus the simulated point cloud (bottom) of that same mission. Both the histogram (top) and simulation verify the point density, and the dark bars in the histogram correctly indicate the presence of point clustering.

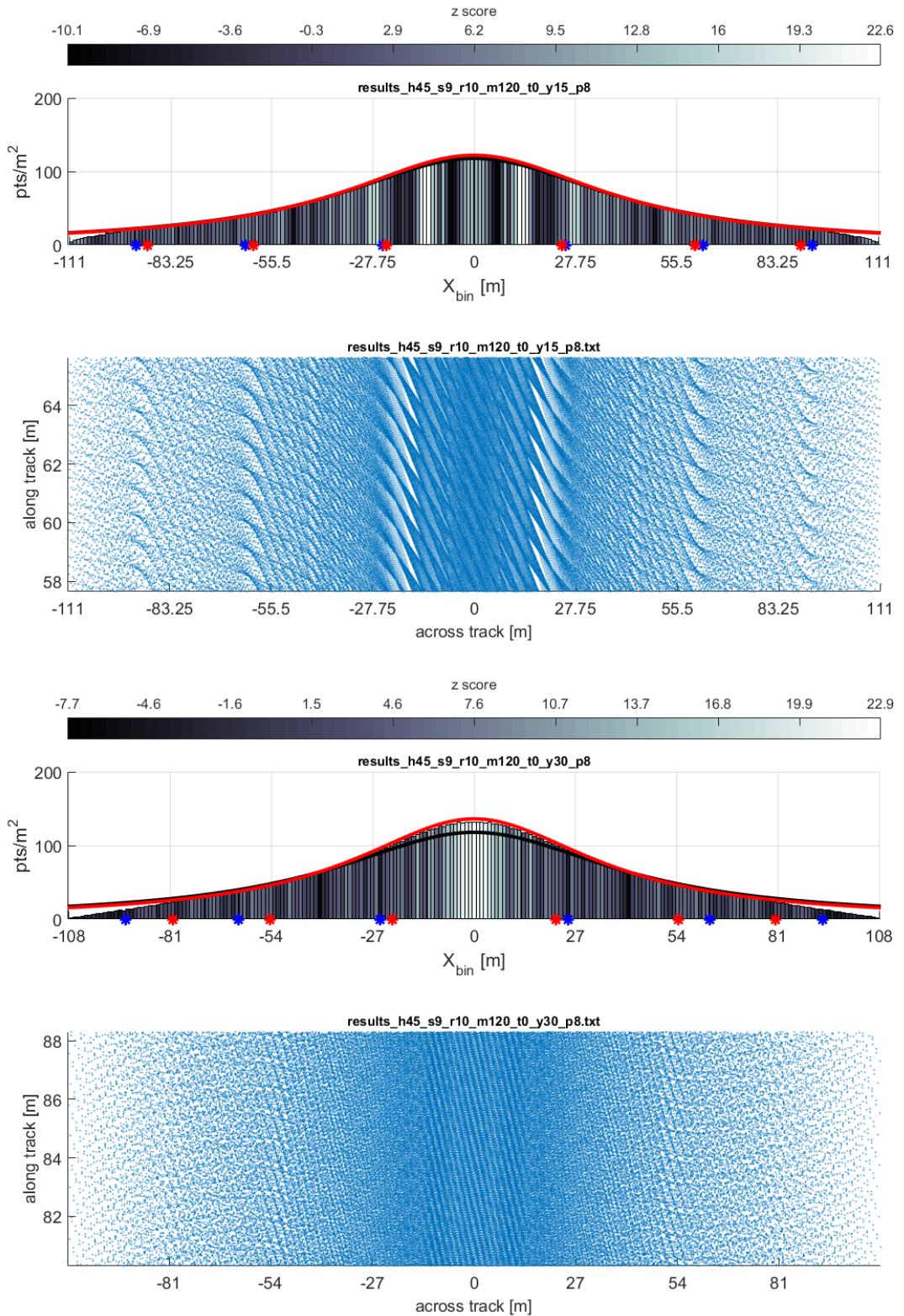


Figure 2-12. Effects of yaw on point density and gap locations. The results of the original PDF and gap equation are plotted in black and blue, respectively. The results from the generalized PDF and gap equations are plotted in red.

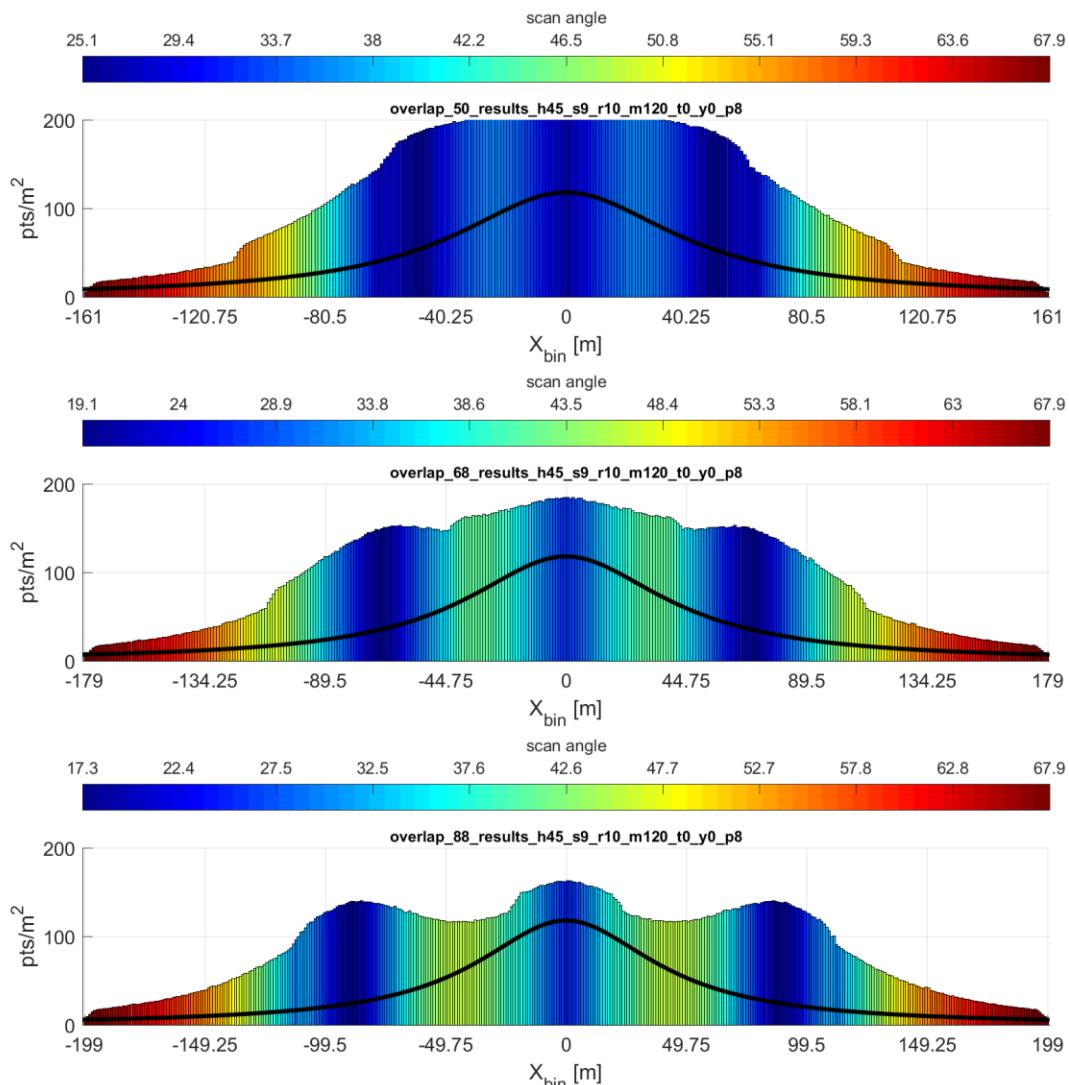


Figure 2-12. Average scan angle. Flying height 45 m, 9 m/s forward velocity, with parallel flight lines spaced at (top to bottom) 50, 68, and 88 m.

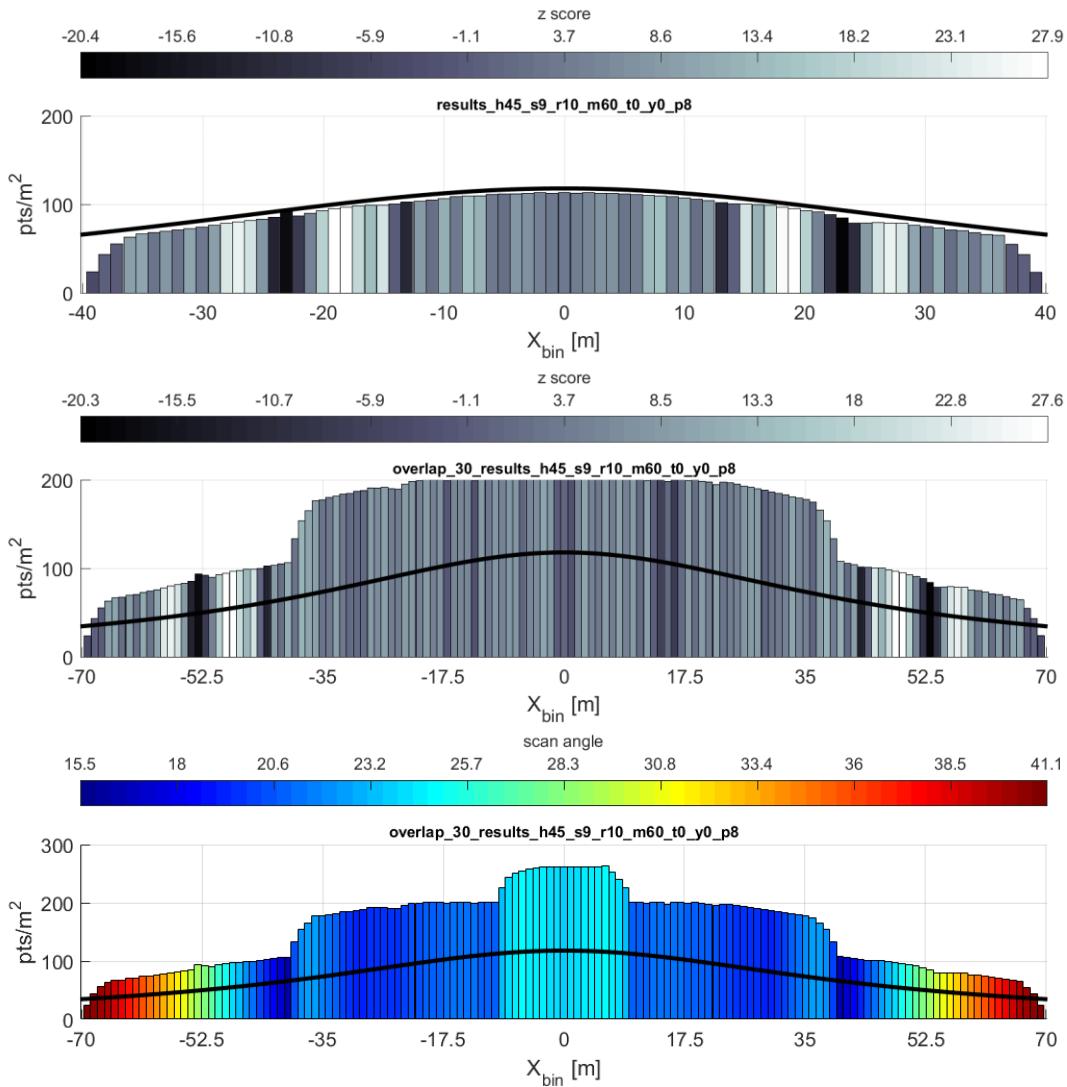


Figure 2-13. Mission planning for artificially limited maximum range. The most noted effect of limiting the maximum range is the drastically decreased profile width. TOP: single flight lines, flying height 45 m, forward speed 9 m/s, max range 60 m. MIDDLE: Overlapping flight lines, 30 m spacing, colored by nearest neighbor z-score. BOTTOM: Overlapping flight lines colored by average scan angle

Table 2-1. Gap equation results, rotation rate 5 Hz.

h [m]	v <sub>z</sub> [m/s]	Lateral distance x [m] at integer <i>i</i>				
		1	2	3	4	5
30	4	cplx	34.6	61.8	86.6	-
30	9	41.9	98.6	-	-	-
30	15	80.5	-	-	-	-
45	4	cplx	8.6	51.9	79.8	-
45	9	25.1	92.8	-	-	-
45	15	73.2	-	-	-	-
60	4	cplx	cplx	33.5	69.3	97.6
60	9	cplx	83.8	-	-	-
60	15	61.5	-	-	-	-

Note: Gap values greater than 100 m have been excluded from these results. Complex results are labeled as "cplx."

Table 2-2. Gap equation results, rotation rate 10 Hz.

h	v <sub>z</sub>	Lateral distance x [m] at integer <i>i</i>									
		1	2	3	4	5	6	7	8	9	10
30	4	cplx	cplx	16.8	34.6	48.8	61.8	74.4	86.6	98.6	-
30	9	cplx	41.9	71.3	98.6	-	-	-	-	-	-
30	15	30.7	80.5	-	-	-	-	-	-	-	-
45	4	cplx	cplx	cplx	8.6	35.4	51.9	66.4	79.8	92.8	-
45	9	cplx	25.1	62.9	92.8	-	-	-	-	-	-
45	15	cplx	73.2	-	-	-	-	-	-	-	-
60	4	cplx	cplx	cplx	cplx	cplx	33.5	53.2	69.3	83.8	97.6
60	9	cplx	cplx	48.8	83.8	-	-	-	-	-	-
60	15	cplx	61.5	-	-	-	-	-	-	-	-

Table 2-3. Gap equation results, rotation rate 20 Hz.

h	v <sub>z</sub>	Lateral distance x [m] at integer i									
		1	2	3	4	5	6	7	8	9	10
30	4	cplx	cplx	cplx	cplx	cplx	16.8	26.6	34.6	41.9	48.8
30	9	cplx	cplx	24.4	41.9	57.0	71.3	85.1	98.6	-	-
30	15	cplx	30.7	57.0	80.5	-	-	-	-	-	-
45	4	cplx	cplx	cplx	cplx	cplx	Cplx	cplx	8.6	25.1	35.4
45	9	cplx	cplx	cplx	25.1	46.1	62.9	78.2	92.8	-	-
45	15	cplx	cplx	46.1	73.2	97.5	-	-	-	-	-
60	4	cplx	cplx	cplx	cplx	cplx	Cplx	cplx	cplx	cplx	cplx
60	9	cplx	cplx	cplx	cplx	23.5	48.8	67.4	83.8	99.3	-
60	15	cplx	cplx		61.5	89.1	-	-	-	-	-

Table 2-3. Continued.

h	v <sub>z</sub>	Lateral distance x [m] at integer i									
		11	12	13	14	15	16	17	18	19	20
30	4	55.4	61.8	68.1	74.4	80.5	86.6	92.6	98.6	-	-
30	9	-	-	-	-	-	-	-	-	-	-
30	15	-	-	-	-	-	-	-	-	-	-
45	4	44.1	51.9	59.3	66.4	73.2	79.8	86.3	92.8	99.1	-
45	9	-	-	-	-	-	-	-	-	-	-
45	15	-	-	-	-	-	-	-	-	-	-
60	4	19.2	33.5	44.1	53.2	61.5	69.3	76.7	83.8	90.8	97.6
60	9	-	-	-	-	-	-	-	-	-	-
60	15	-	-	-	-	-	-	-	-	-	-

## CHAPTER 3

### OBLIQUE UAS PHOTOGRAMMETRY IN FORESTED SCENES

#### **Introduction**

Photointerpretation and photogrammetry have been used for forest inventory dating back to at least 1929, when aerial timber cruises were conducted in Canada (Spurr, 1948). Stereo photography for forestry applications has been in use since at least 1958 (Avery, 1958). In fact, at least as early as 1967 photogrammetric methods of forest sampling utilizing cameras mounted to a helicopter had been shown to produce results that were not statistically different from conventional ground-based methods. Tree species, tree height, and crown width were determined from the photos, parameters which were then used to develop equations for single-tree volume and diameter at breast height, or DBH (Lyons, 1967). In the 1990s the advent of digital photogrammetry led to further applications in forest inventory, e.g. determining canopy height (Gagnon et al., 1993).

Even as the process for gathering 3D data from stereo photography shifted to the digital realm, the spatial resolution of that data was still limited. The collection of data points, i.e. the 3D locations of physical features, was performed manually by an operator. This manual process prohibited the dense 3D reconstruction of the forest. The operator, limited by time, could only select a handful of points of interest to estimate forest metrics.

#### **Automated 3D Reconstruction of the Forested Scene**

A powerful tool in digital photogrammetry today is structure-from-motion (SfM), method of generating accurate, high-resolution topographic datasets from nonmetric photographs (Westoby et al. 2012). Though similar in many ways to conventional photogrammetry, SfM offers a lower capital cost and more automated processing than conventional photogrammetric workflows. SfM, using low-cost, commercial, off-the-shelf (COTS) cameras, can be used to

produce a high-resolution 3D reconstruction (often referred to as a point cloud) of the photographed area.

The ability to use common digital photos to obtain the 3D structure of objects has led to increased interest in 3D mapping in many fields, particularly forestry and forest ecology. Digital cameras are lightweight, inexpensive, and simple to deploy, and the SfM methods used to process the photos have been shown to be comparable to light detection and ranging (lidar) in producing topographic datasets (Westoby et al., 2012; Fonstad et al., 2013).

Photogrammetry from an unmanned aerial system (UAS) platform allows for not only a high spatial resolution, but also a high temporal resolution. UAS are much easier to deploy than a conventional aircraft, which allows for more frequent data collection. This high temporal resolution can be greatly beneficial for applications such as stress monitoring or plant phenotyping for forest tree breeding (Dhondt et al., 2013).

### **Feature Matching in a Forested Scene**

Previous studies have demonstrated the successes of SfM reconstruction of open, non-vegetated areas of various terrains, as cited above. In a mission with a low base-to-height ratio—i.e., the average distance between exposure stations is small when compared to the range of the target from the cameras—a particular feature may be visible in five or more photos. Solving for the 3D location of the feature becomes more robust, assuming the errors in the location of said feature in each photo are random. By contrast, the heterogeneous structure of forests, with its sharp gradients in height, varying textures, and irregular shapes, presents a challenge to the SfM workflow, particularly the feature matching process. A particular feature may appear in the minimum of two photos; any error in the feature’s location on either photo can lead to a large error in the solution for that feature’s 3D location. In the experience of the author, this is the likely cause of “false positives” in the 3D point clouds of forested scenes generated from low-

altitude nadir photography (Figure 3-1). (Too few feature matches in certain areas is also the likely cause for false negatives, or missing information in the 3D point cloud where photo coverage exists.)

Some error in the stereo parallax—the change in apparent position of a feature from one stereo image to the next—will lead to a miscalculation of its range from the cameras. Further evidence to support this supposition is the orientation of clusters of false positive points. Usually they are longer than they are wide in the direction of the optical axes of the cameras; as implied in Figure 3-2, these inaccuracies would likely take on such a shape. Further inaccuracies in the reconstruction of the scene may be caused by this poor geometry between exposure stations, but may not be easily spotted by inspecting the point cloud.

### **Dynamic, Close-range Photogrammetry**

The geometry of the photos used to reconstruct a scene plays a central role in the accuracy of the scene. Current methods resemble a scaled-down version of a conventional aerial photogrammetry mission flown from high altitude (typically >1000 m) with a manned aircraft. Such a conventional mission uses nadir-looking photography. The crucial difference between a conventional mission and a UAS mission is the ratio of target height to flying height. A conventional mission flown from 1000 m over an area where the average height of the targets (buildings, for example) is 20 m has a target-to-flying height ratio of 1:50, while a UAS mission flown at 75 m over a forested scene with an average of 15-m-tall trees has a target-to-flying height ratio of 1:5. In other words, the targets are much closer to the camera; small changes in camera position during photography can lead to a greater change in apparent position of a feature from one photo to the next. Two key elements of photogrammetry are at play in this scenario: the high apparent change in position (or parallax) between the features may lead to decreased error in the computed positions of the feature, but the increased relief displacement between two key

features—say, the top and bottom of a tree stem—adds an element of distortion to each photo (Wolf et al. 2014). In cases presenting a high target-to-flying height ratio, a UAS photogrammetry mission should not be treated as low-altitude aerial photogrammetry, but rather as dynamic, close-range photogrammetry.

The most stated difference between conventional aerial and close-range photogrammetry is the geometry of the cameras (herein referred to as exposure stations). The converging optical axes of the exposure stations provides a stronger geometry for both the bundle adjustment and 3D reconstruction of the scene. The demand for aerial oblique imagery in commercial applications appears to be growing as well, as indicated by recent patents filed by Google (Reece 2014) and Pictometry International (Schultz et al., 2010; Schultz et al., 2011) pertaining to the collection and processing of oblique imagery. The resulting products from Google's oblique imagery, collected from manned aircraft at higher altitudes, have been quite impressive, as evidenced by the dense and accurate 3D reconstruction of buildings and vegetation (in some cases, individual trees!) in the urban areas where oblique images have been collected. A number of studies using low-altitude and/or UAS oblique imagery has been noticed in the literature, mostly focused on reconstruction of structures or for monitoring geological (Aicardi et al., 2016, James & Robson, 2014; Nex & Remondino, 2014). To the author's knowledge, only one recent study of the sort explored the reconstruction and mensuration of a forested scene using low-altitude oblique images collected via UAS, demonstrating moderate success in reconstructing and mensurating the tree stems (Fritz et al., 2013).

This study explores the process of planning UAS photogrammetric missions in forested scenes to more closely align with the principles of close-range photogrammetry with the goal of achieving a more accurate 3D reconstruction of the forested scene, particularly the forest canopy.

First, a quantitative analysis is presented comparing nadir and oblique imagery over a relatively flat and open (unforested) scene in which the errors of 20 checkpoints are assessed. This analysis serves as a baseline for expected reconstruction differences between nadir and oblique imagery and comparing the reprojection. Next, a qualitative analysis is conducted using the canopy a set of three canopy height models (CHMs) of a planted pine stand generated from nadir imagery, oblique imagery, and lidar, all from the UAS platform. An application analysis is also presented in which canopy-based individual tree detection (ITD) is conducted on the three CHMs and the results are compared with the a complete timber cruise, with the goal of assessing the utility of the CHMs for a forestry application. Together, these analyses are designed to test the notion that oblique imagery will lead to a more accurate 3D reconstruction of a heterogeneous, forested scene than will nadir imagery.

## Methods

### Study Areas

**Quantitative study, Jonesville.** RGB images from both the nadir and oblique poses were collected 15 June 2018 at Jonesville Park north of Gainesville, Florida (Figure 3-3). The site (“Jonesville”) is an open grass field, roughly 100-by-70 m. The images were collected from the DJI S1000 vertical takeoff & landing (VTOL) UAS, equipped with a gimbal mount that allows for orienting the payload at a predefined tilt angle of 30°. (The process by which this angle was chosen is detailed below.) The camera used is a Canon EOS-Rebel SL1, a commercial, off-the-shelf (COTS) DSLR camera, with a 24-mm lens. Airborne control (position and attitude) was acquired via the Xsens Mt-G-700 GNSS/INS. Five ground control points (GCPs) and 20 checkpoints were placed in the scene, and their coordinates were measured via RTK GNSS. The data collected at this study site were used to perform a quantitative analysis on the reported error

of the checkpoints, to determine if either camera configuration results in a significantly more accurate reconstruction of the scene.

**Qualitative and application studies, Millhopper.** RGB images were collected from both the nadir and oblique poses on 14 March 2018 at the University of Florida Agricultural Experimental Station in the Millhopper neighborhood just north of Gainesville, Florida (Figure 3-4). The specific study site (herein referred to as “Millhopper”) is a 160-by-160-meter plot of pines planted in 2010. The UAS platform and payload were the same as described above. Five GCPs were also placed in the scene, and their coordinates were measured via RTK GNSS. UAS lidar data was also collected over this site on the same day from a second DJI S1000 equipped with the lidar payload described in Chapter 1. These three datasets were used to generate three canopy height models (CHMs) which are compared amongst each other, holding the lidar CHM as control. Specific information on the PhotoScan processing of the three flights can be found in Tables 3-1 and 3-2.

## Mission Simulation

Battery life is a concern with VTOL UAS missions, especially with the generally narrower flight line spacing required for taking images with a DSLR camera. To maximize the efficiency of image data collection, mission planning equations and simulation software were developed for this study. These tools were used to calculate stereo overlapping coverage of photos in the direction of flight (end lap) and in adjacent flight lines (side lap) as well as ground sample distance (GSD), conduct trials for optimal flying heights, and determine at which angle from nadir the camera should be tilted to achieve the hypothesized result of more accurate 3D reconstruction of the scene. This mission simulation also served to verify the planning equations’ results. The mission simulation software was written in the MATLAB programming language. The overall goal was to simulate images taken of a scene at various angles of tilt, and to use

those simulated images to perform a bundle adjustment at various angles of tilt. The bundle adjustment provides statistics on the precision and accuracy of the adjustment, allowing for comparison among the simulations' respective bundle adjustments.

**End lap and side lap.** The MATLAB simulation software accepts as input the following mission parameters: (1) end lap, (2) side lap, (3) flying height, and (4) tilt. The standard methods for calculating the distance between exposure stations (i.e. air base, a function of end lap) and the distance between adjacent flight lines (i.e. air width, a function of side lap) assumes vertical (nadir) photography; therefore, new equations needed to be derived for calculating these parameters.

The air base  $B$  for oblique images can be found by

$$B = H' \left[ \tan\left(t + \frac{\phi}{2}\right) - \tan\left(t - \frac{\phi}{2}\right) \right] (1 - E) \quad (3-1)$$

where  $H'$  is the flying height,  $t$  is the angle of tilt,  $\phi$  is the field of view of the camera in the direction of flight, and  $E$  is the desired end lap expressed as a percentage (0 – 1). The air width  $W$  can be expressed as

$$W = H' \frac{w}{f \cos t} (1 - S) \quad (3-2)$$

where  $w$  is the width of the camera's format,  $f$  is the camera's focal length, and  $S$  is the desired side lap expressed as a percentage (0 – 1). The derivations for Equations 3-1 and 3-2 can be found in Appendix C.

**Camera simulation.** The simulation software also requires two variables about the camera to be defined: focal length and format size. The parameters for the simulation of the camera are those of the Canon EOS Rebel SL1, which is the camera used for data collection. The software simulates the collection of three parallel strips of four images each, given the four mission parameters listed above. These simulated images are, to be more precise, arrays of 2D

points which are themselves back projections of a  $3 \times 3 \times 3$  array of 3D points onto planes at the positions and angular orientations specified by the mission parameters (Figure 3-5). For this simulation, the camera was assumed to be an ideal pinhole camera; no elements of lens distortion were added. Random noise of 0.5 pixel is added to each back projection in both the x- and y-axes in image space to simulate the random error in the detected location of some feature in an image. Adding this noise turns out to be a crucial step in simulating a bundle adjustment, however; without the addition of this random noise, the residuals of the bundle adjustment would be zero, leading to an error variance of zero. Further explanation of how the images are simulated can be found in Appendix D.

## Mission Planning

For both study sites, two missions were flown over the scene: (1) a “conventional” mission collecting nadir photos, and (2) a mission collecting oblique photos, with the camera inclined at some tilt angle from nadir. The tilt angle of  $30^\circ$  was determined after considering both the simulated accuracy assessments and the changes in the length of flight lines caused by covering a study area with an oblique, as opposed to nadir, camera. (Further discussion of these considerations can be found in the Results section below.)

Both nadir and oblique missions were controlled with respect to percent end lap, percent side lap, and average range to target. The shutter of the camera aboard the UAS is fired on a timer at a regular interval  $t_{\text{shutter}}$ , so to control for end lap, the forward speed of the camera was manipulated. Knowing the necessary air base  $B$  for the required end lap as described above, the forward speed  $v$  is determined by  $v = B/t_{\text{shutter}}$ . Controlling for side lap was done by adjusting the flight pattern such that adjacent flight lines are the appropriate distance  $W$  apart. Average

range to target  $d_{\text{target}}$  can be defined approximately as the distance between the camera and the target surface, and is a function of the flying height  $H'$  and tilt of the camera  $t$ :  $d_{\text{target}} = H' \cos t$ .

Another consideration with oblique photos is the ground sample distance (GSD) of each pixel of a photo and how it relates both to the desired resolution of the photos—rarely an issue at low altitudes—and the shutter speed of the camera. With a camera in motion, best practice is to ensure that the camera travels no farther than the length of 1 GSD while the shutter is open. Thus, the maximum forward speed of the camera  $v_{\max}$  is limited by (1) GSD in the direction of flight—a function of flying height and pixel size—and (2) shutter speed. With vertical photography, the GSD is constant, and can be found by multiplying the photo scale by the pixel size. The GSD is variable in oblique photography, growing larger in the direction of tilt. The GSD in the direction of flight that corresponds to a column  $p$  of pixels of the photo is expressed by

$$\text{GSD}_{x,p} = H' [\tan(t + \mu_{p+1}) - \tan(t + \mu_p)] \quad (3-3)$$

where  $\mu_p$  is the angle formed between the optical axis of the camera and the ray from the perspective center of the camera to the pixel in column  $p$ , with  $m$  being the number of columns of pixels in the format. The derivation of Equation 3-3 can be found in Appendix C.

Both nadir and oblique missions were flown in a “serpentine” flight pattern (Figures 3-3, 4). For the oblique photo mission, this flight pattern lends to the “stronger” geometry of converging photos (or, more precisely, the converging of the optical axes of the camera exposure stations), whereas the nadir mission exhibits the “weaker” geometry of non-converging, parallel optical axes. To ensure full coverage of the study site, however, the flight lines for the oblique mission must be lengthened. This extra length is a function of the tangent of the tilt angle, or  $\ell =$

$H' \tan t$ . Also, for both study sites, the missions utilize the same GNSS/INS payload for airborne control and the same ground control points.

## Reconstruction Analysis

For the qualitative analysis of the Millhopper data, two CHMs were created, one from each camera configuration. Agisoft PhotoScan was used to create the point cloud from which the CHMs were created. The workflow in PhotoScan to create the 3D reconstruction of the scenes—i.e., dense matching point clouds—is based on the process presented by the USGS National UAS Project Office (2017):

1. Align photos (high accuracy, 4,000 key point limit, no tie point limit)
2. Optimize cameras (focal length  $f$ , principal point offsets  $c_x, c_y$ , radial distortion coefficients  $k_1, k_2, k_3$  and decentering distortion coefficients  $p_1, p_2$ )
3. Selectively remove tie points with greatest reconstruction uncertainty, optimizing the cameras after each set of points is removed
4. Dense point cloud reconstruction (medium quality, depth filtering disabled)
5. Classify point cloud to find “low noise,” an artifact in dense reconstruction that presents itself as false negatives below the ground
6. Export dense cloud without low noise

After the point clouds are created, the CHMs were made using LAStools (Isenburg, 2012) using the “pit-free CHM” process presented by Khosravipour et al. (2014):

1. Normalize the height of the point cloud—i.e., “flatten” the ground such that the z-coordinate of all points is replaced by height above ground
2. Create a digital elevation model (DEM) of the area
3. Thin the point cloud one half of the cell size of the desired CHM—in this case, 0.16667 m thinning step size for a CHM with cell size 0.33333 m
4. From the thinned point cloud, create CHMs with all points above 0 m, 2 m, 5 m, 10 m, and 15 m (all heights above ground)
5. Merge the temporary CHMs into a single CHM, keeping for each cell the highest value

This process was followed for both the nadir and oblique point clouds as well as the lidar point cloud, resulting in the three CHMs to be used for the qualitative analysis. Using MATLAB, the CHMs were then subtracted from each other in three configurations:

1. Nadir CHM minus lidar CHM
2. Oblique CHM minus lidar CHM
3. Nadir CHM minus oblique CHM

For the quantitative analysis of the Jonesville data, only the first four steps of the PhotoScan workflow shown above were used, except that for the fourth step, a lower-quality dense cloud was created. The dense cloud was used to identify in the scene the locations of the checkpoints. Once identified in the scene, the approximate locations of the checkpoints were back-projected into each image in which they appeared. The locations of the checkpoints were then manually adjusted in each photo. The reprojected error for each control point was reported by PhotoScan and used for the subsequent analysis.

## Results

### Mission Simulation and Planning

Compared to parallel optical axes—which is the case for nadir photography—converging optical axes of the camera exposure stations provide stronger geometry for the intersection of the rays of light traced from an object to the lens, with the strongest geometry occurring at an angle of intersection of  $90^\circ$ . With a tilt angle of  $45^\circ$ , the intersections of light rays of the targets in the scene will be much nearer to  $90^\circ$  in converging pairs of photos. This expected increase in accuracy is supported by the bundle adjustment results presented in Figure 3-6.

The change in GSD in the direction of flight as tilt increases is shown in Figure 3-7. At a tile of  $45^\circ$ , a drastic increase in GSD presents itself along the format in the direction of flight. It becomes apparent that the desire for a more accurate bundle adjustment (and, assumedly, a more accurate reconstruction of the scene) must be balanced with the need for a relatively small GSD.

This balance should be not difficult to achieve, as the increased accuracies of the bundle adjustments begin to level off around 30° degrees of tilt; at the arbitrarily-chosen flying height of 80 m, 30° of tilt does not present an unreasonable difference between the GSDs at either end of the format.

### **Quantitative Checkpoint Analysis**

The results for the checkpoint analysis are detailed in Tables 3-3:5. The nadir and oblique reprojected checkpoint errors first underwent a Shapiro-Wilk test to test for the assumption of normal distribution of the errors within sample. The nadir and oblique samples returned *p*-values of 0.17 and 0.68, respectively; for both samples, the null hypothesis of normality is not rejected. Under the assumption of normality, further statistical testing could proceed. Because the variances for the two samples were quite different ( $\sigma_n^2 = 4.76 \text{ cm}$ ,  $\sigma_o^2 = 0.92 \text{ cm}$ ), an F-test was performed, which verified the variances were significantly different ( $p = 0.0004$ ). Thus, the final test is the two-sample *t*-test assuming unequal variances, also known as Welch's *t*-test. The null hypothesis of equivalence is rejected in favor of the alternative hypothesis that the mean of the nadir errors is greater than the mean of the oblique errors ( $p = 0.0036$ ).

### **Qualitative CHM Analysis**

At the stand level, the oblique imagery appears to produce a more accurate reconstruction of the scene when compared to the nadir imagery (Figures 3-8 and 3-10). This is further corroborated by comparing the two differenced CHMs comparing nadir and oblique each to lidar, as shown in Figures 3-9 and 3-11. Summary statistics for all five of the CHMs can be found in Table 3-6. The nadir imagery appears to produce “false negatives” at the perimeter of tree canopies. However, closer inspection reveals that the oblique imagery may be encountering reconstruction errors of its own. This is further discussed in the Conclusion.

### **Application: Individual Tree Detection (ITD)**

(To determine the effectiveness of using oblique imagery in a common forestry application, the following concurrent work by Luiz Ramalho de Oliveira is presented here.)

#### **Methods**

Using the rLIDAR package in the R programming language (Silva et al., 2017), each CHM was smoothed by a mean filter under various smoothing window sizes (SWS). The rLIDAR package identifies individual trees in a smoothed CHM using local maxima detection, in which the CHM is searched for local maxima using a fixed window size (FWS). The accuracy of the ITD was evaluated in terms of true positive (TP, correct detection), false negative (FN, omission error) and false positive (FP, commission error). From these values, the recall ( $r$ , tree detection rate), precision ( $p$ , correctness of the detected trees) and  $F$ -score ( $F$ , overall accuracy), as explained by Li et al. (2012), are calculated using the following equations (Mohan et al., 2017):

$$r = \frac{TP}{TP+FN}$$

(3-4)

$$p = \frac{TP}{TP+FP}$$

(3-5)

$$F = 2 \left( \frac{r \times p}{r+p} \right)$$

(3-6)

Various SWS and FWS were used to best accommodate the various stem spacings found in the stand, which is noted in the results. Both the SWS and FWS windows sizes were optimized for each subplot within the study area in an effort to maximize the  $F$ -score. The subplots are planted at two different spacings, Narrow (N), 1.83 x 3.65 m, and Wide (W), 3.65 x 3.65 m. These plots

are also either thinned ( $T = 1$ ) or not thinned ( $T = 0$ ). There is additionally an independent plot called Biomass (B)—(3.65 x 1.83 m, no thinning).

## Results

The complete results of ITD in the three datasets if found in Tables 3-7:9. When compared to ITD in the lidar CHM ( $F = 0.98$ ), ITD in the nadir CHM ( $F = 0.97$ ) performs notably better than ITD in the oblique CHM ( $F = 0.93$ ). The performance of ITD in both of the datasets is similar except for one striking difference: only 52 false positives are detected in the nadir CHM as compared to 119 in the oblique CHM. The ITD in the nadir CHM thus garners a higher  $p$  score, or correctness of detected trees, which makes the difference between the two datasets. Further discussion of these results can be found in the Conclusion.



Figure 3-1. Artifacts in dense matching point clouds. “False positives” in the 3D point clouds of forested scenes generated from low-altitude nadir photography. The false positives manifest both above and below ground. Figure courtesy of Dr. Ben Wilkinson, University of Florida.

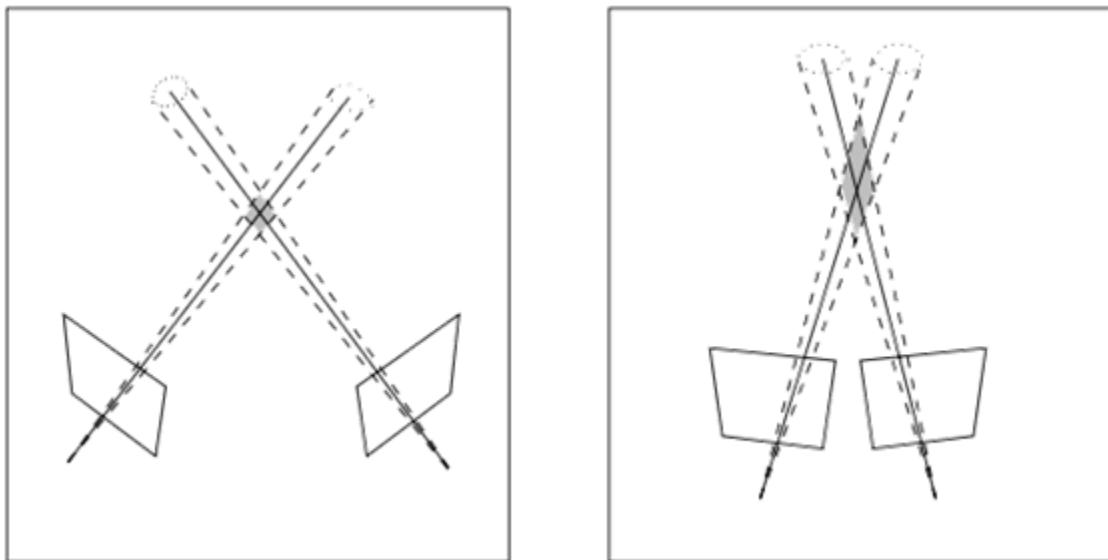


Figure 3-2. Geometry of exposure stations. The geometry of exposure stations impacts the accuracy of the reconstructed scene, especially in the dimension parallel to the optical axes. The possible error of the reconstructed point is reflected in the intersection of the two cones of uncertainty, colored gray (Hartley and Zisserman, 2012).



Figure 3-3. Jonesville Park study site with mission plans. Flight lines are shown for the nadir (yellow) and oblique (magenta) missions. GCPs (triangle) and checkpoints (diamond) are shown as well. Background image courtesy of Google.



Figure 3-4. University of Florida Agricultural Experimental Station with mission plans. Referred to also as the Millhopper site, the flight lines are shown for the lidar (cyan), nadir (yellow), and oblique (magenta) missions. GCPs shown as triangles. Background image courtesy of Google.

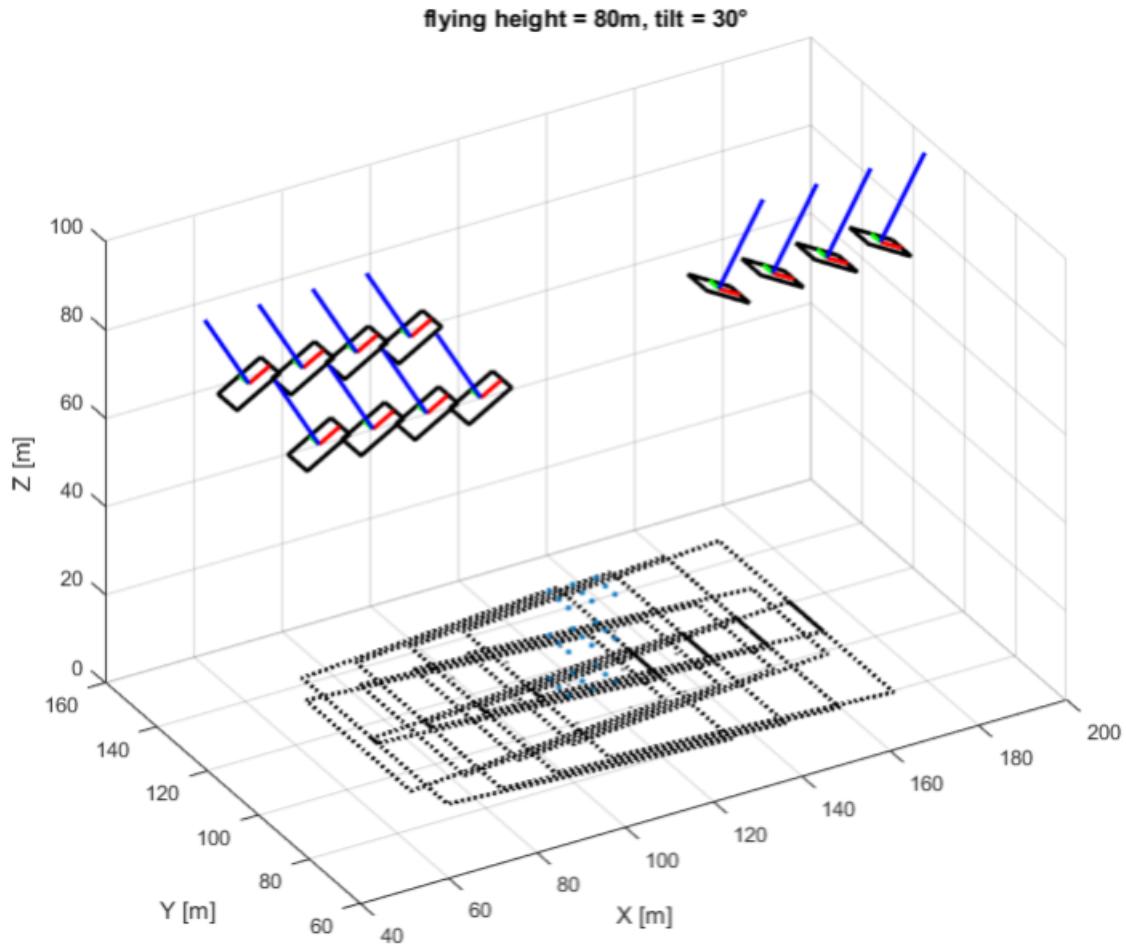


Figure 3-5. Oblique mission simulation. This diagram depicts a simulated mission generated by the MATLAB simulation software. The exposure stations are shown along the top, the array of 3D points are shown in the middle, and the footprints of the exposure stations are shown in dotted lines along the bottom.

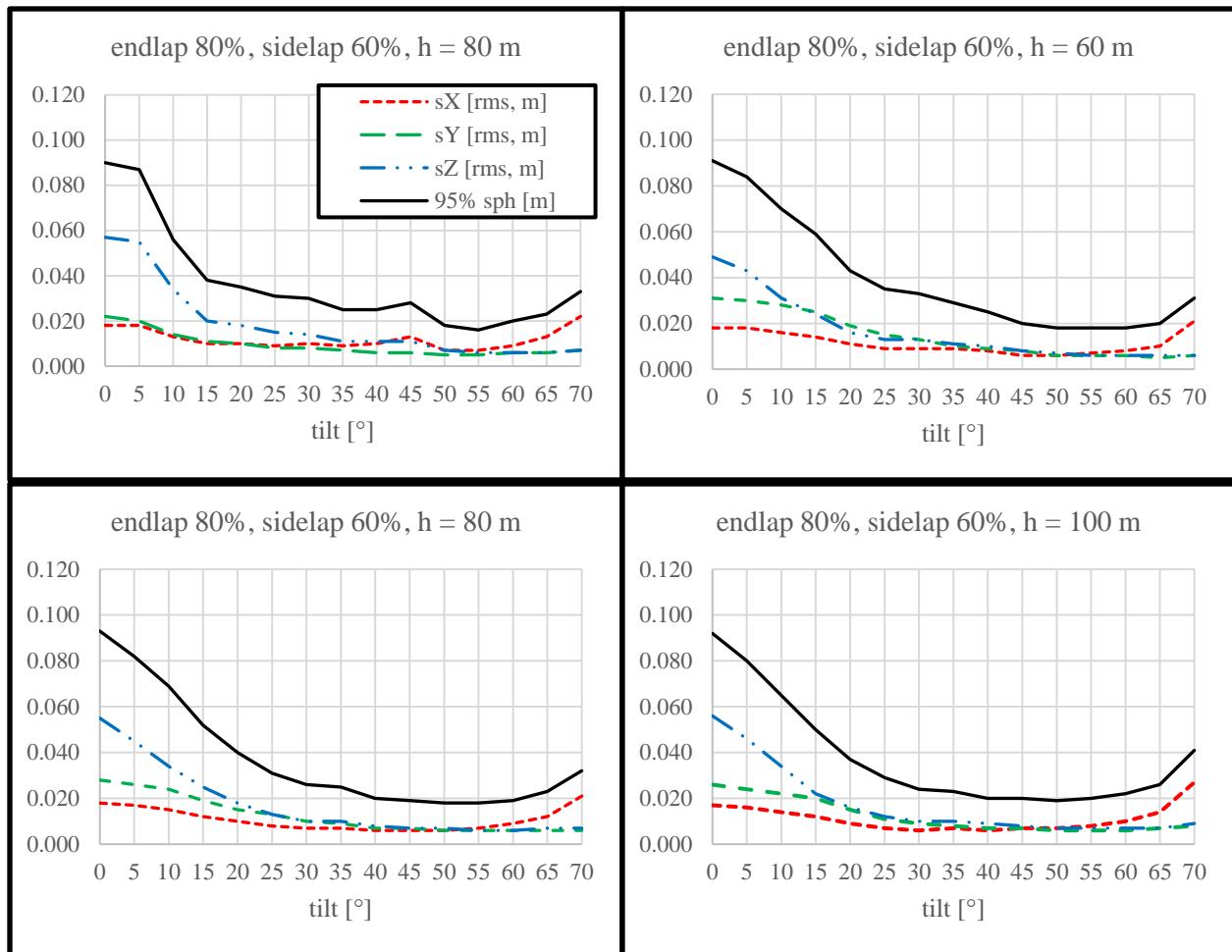


Figure 3-6. Simulated bundle adjustment results.

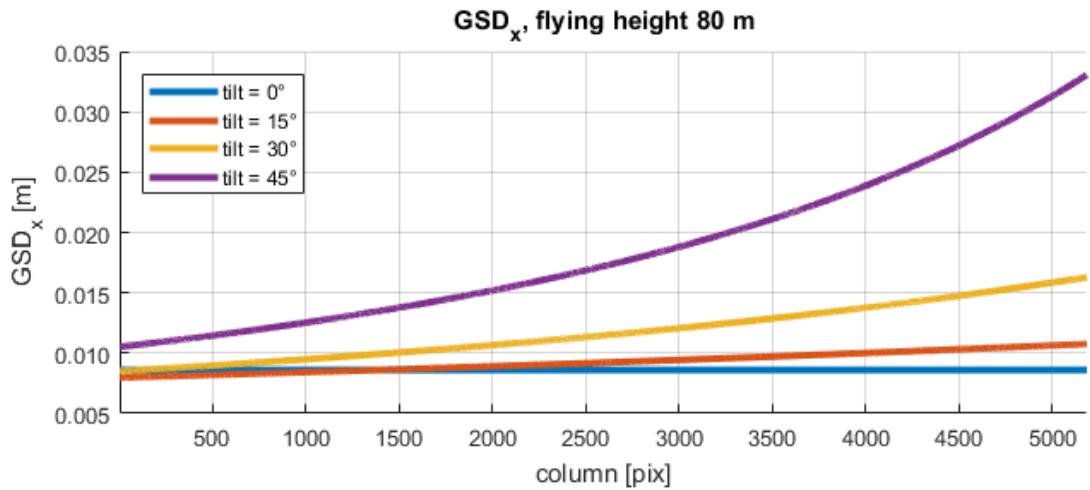


Figure 3-7. Ground sample distance along format as a function of tilt.

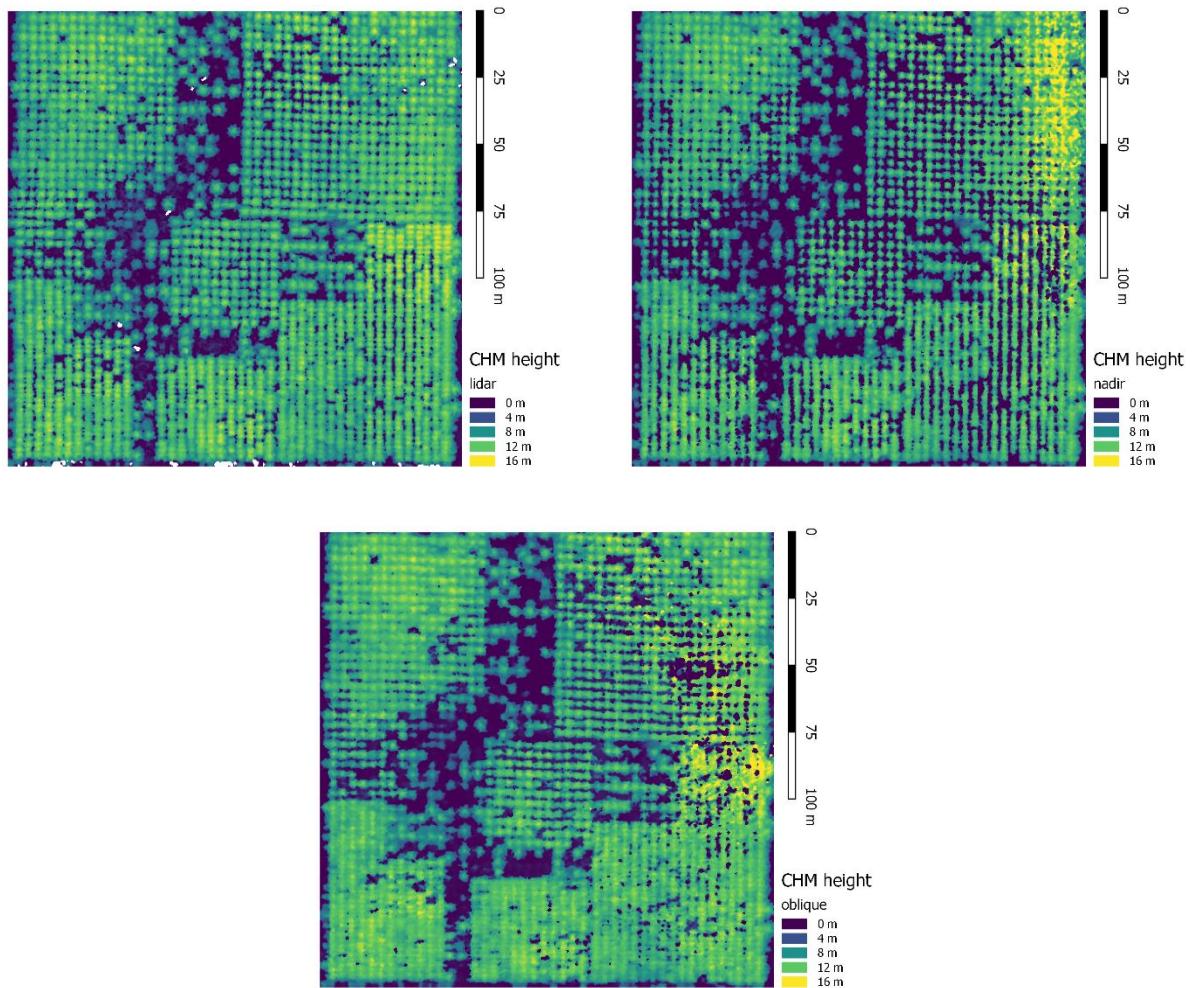


Figure 3-8. Canopy height models, Millhopper study site. Clockwise from top left: lidar, nadir imagery, and oblique imagery.

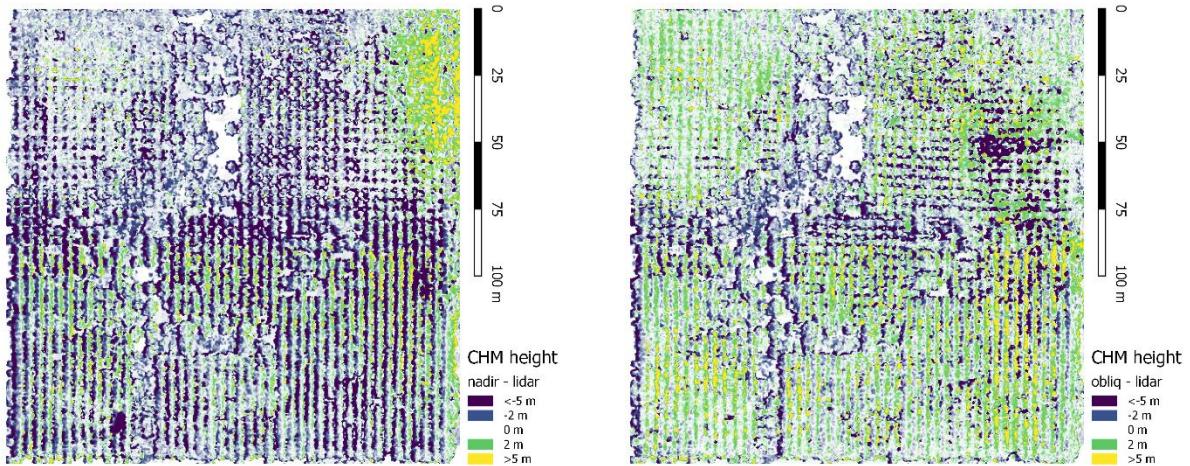


Figure 3-9. Differenced canopy height models, Millhopper study site. The summary statistics for these can be found in Table 3-6.

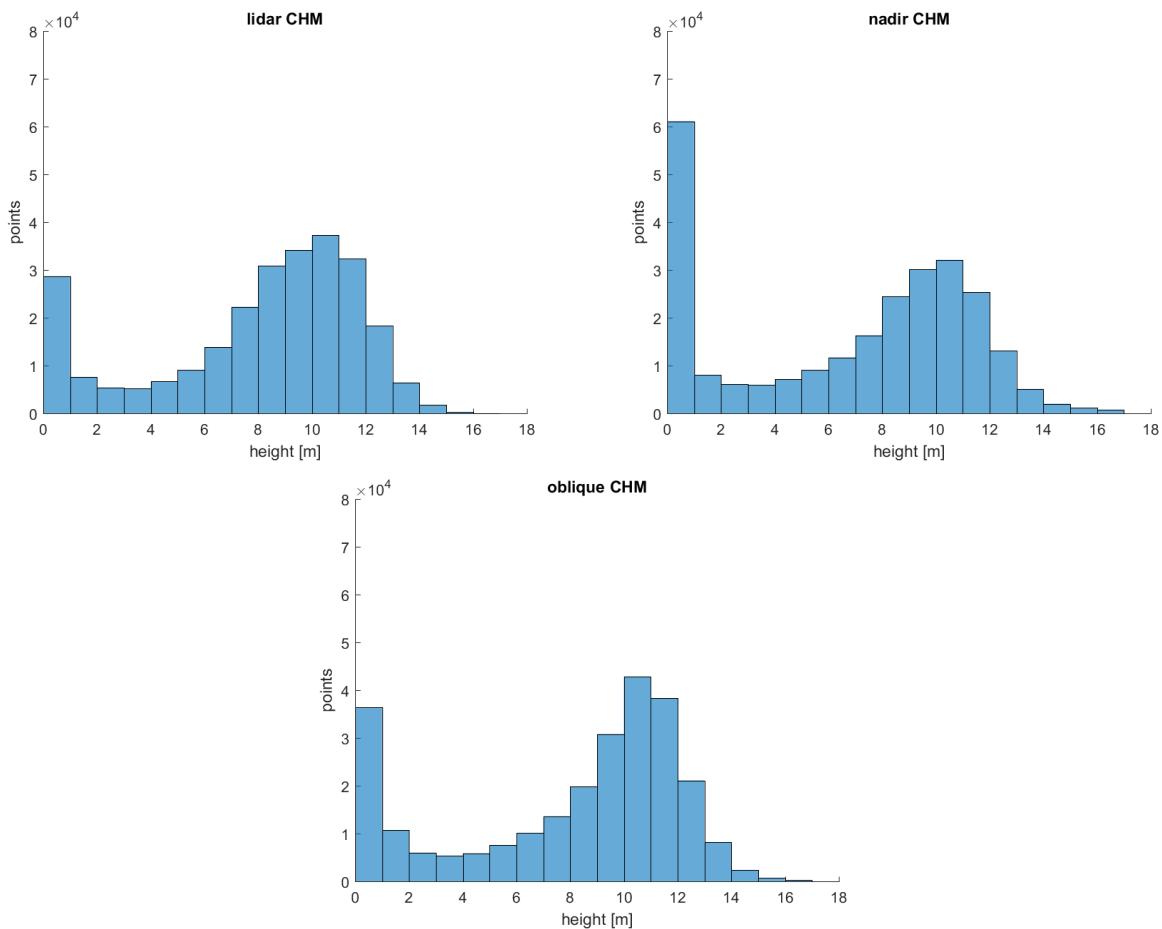


Figure 3-10. Histograms of canopy height models. Clockwise from top left: lidar, nadir imagery, and oblique imagery.

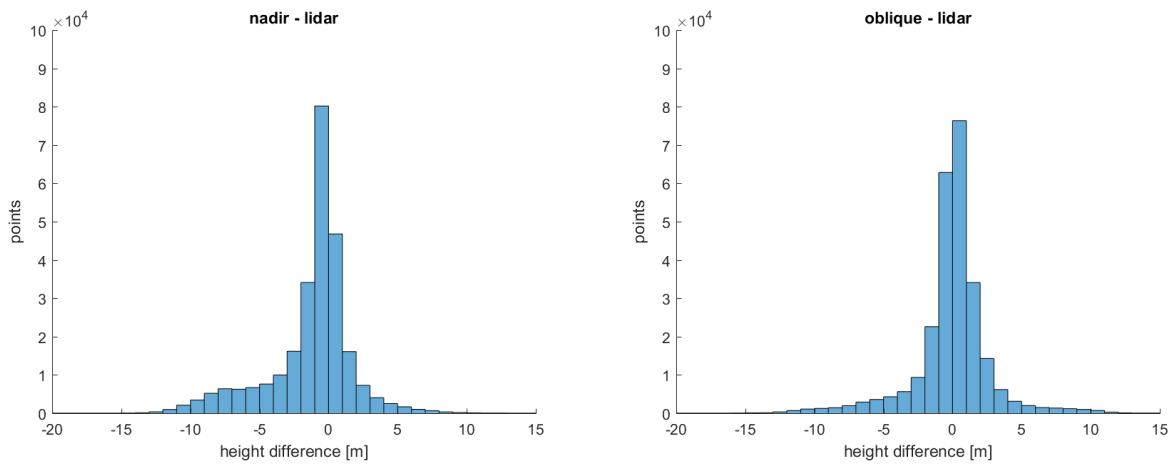


Figure 3-11. Histograms of differenced canopy height models.



Figure 3-12. Distortion in dense matching reconstruction of the scene. In the nadir dense matching point cloud (left), the canopies appear less distorted than in the oblique cloud (right). Direction of flight with respect to each image is left-right.

Table 3-1. PhotoScan processing details.

	Jonesville		Millhopper	
	Nadir	Oblique	Nadir	Oblique
Aligned cameras	128	149	369	502
RMS projection error [pix]	0.348	0.335	0.364	0.401
Max reprojection error [pix]	1.338	0.697	1.387	2.524
Matching time (h:mm:ss)	00:01:52	00:01:33	00:03:19	00:04:24
Alignment time	00:00:57	00:00:47	00:02:27	00:02:54
Depth maps generation time	01:17:00	00:37:51	00:17:17	00:48:19
Dense cloud generation time	00:16:49	00:11:28	00:19:43	01:10:00

Table 3-2. Processing specifications.

Computer model	Dell Precision T3610
CPU	Intel Xeon E5-1620 @ 3.70 GHz (4 cores)
GPU	NVIDIA K4000
RAM	16 GB DDR3
PhotoScan version	1.4.2 build 6205

Table 3-3. Checkpoint error results.

Point	Oblique				Nadir			
	X [cm]	Y [cm]	Z [cm]	Total	X [cm]	Y [cm]	Z [cm]	Total
P1	0.2	0.7	1.5	1.7	0.0	1.5	-2.0	2.5
P2	0.1	1.4	-0.7	1.5	-0.3	1.4	-2.6	3.0
P3	0.0	0.9	-2.9	3.0	0.2	2.1	-3.0	3.6
P4	-0.5	1.5	-2.4	2.9	0.2	1.9	-2.7	3.3
P5	-0.9	0.7	-3.6	3.8	-0.3	0.6	-0.7	1.0
P6	1.6	0.0	-3.3	3.7	1.6	0.4	1.1	1.9
P7	0.4	-1.3	-3.9	4.1	1.1	-1.5	0.5	1.9
P8	1.3	0.8	-3.1	3.5	0.3	0.7	-1.7	1.9
P9	-1.2	-2.1	-3.7	4.4	-1.9	-1.9	-5.1	5.8
P10	-0.1	-0.8	-2.1	2.3	0.2	-0.9	-4.7	4.8
P11	-1.8	-1.7	-3.1	4.0	-2.4	-1.2	-3.8	4.7
P12	0.5	-0.8	-1.6	1.9	-0.8	-0.5	-5.9	5.9
P13	-1.0	-1.5	-1.5	2.4	-1.2	-1.0	-7.8	7.9
P14	-0.6	1.4	-2.6	3.1	-0.5	0.8	-4.8	4.9
P15	0.0	0.3	-3.5	3.5	-1.5	0.2	-8.2	8.3
P16	-0.2	3.0	-2.8	4.1	-1.8	1.6	-6.8	7.2
P17	-0.5	-1.3	-2.1	2.5	-1.0	-1.1	-3.4	3.7
P18	0.8	-0.4	-2.7	2.8	0.7	-1.0	-2.9	3.2
P19	0.0	-0.8	-0.5	0.9	-0.3	-1.5	-1.5	2.1
P20	-1.3	0.2	-2.8	3.0	-0.8	-0.7	-0.7	1.3
RMSE	0.19	0.29	0.60	0.69	0.24	0.28	0.93	1.00
$\sigma$	0.86	1.31	1.30	0.96	1.02	1.28	2.57	2.18

Table 3-4. Checkpoint analysis, *F*-test results.

	Nadir	Oblique
Mean	3.95 m	2.95 m
Variance	4.76 m	0.92 m
Observations	20	20
Degrees of freedom	19	19
<i>F</i>	5.157	
$P(F \leq F_{crit})$ one-tail	0.0004	
$F_{crit}$ one-tail	2.168	

Table 3-5. Checkpoint analysis, Welch's *t*-test results.

	Nadir	Oblique
Mean	3.95 m	2.95 m
Variance	4.76 m	0.92 m
Observations	20	20
Hypothesized mean difference	0	
Degrees of freedom	26	
<i>t</i>	1.88	
$P(t \leq t_{crit})$ one-tail	0.04	
$t_{crit}$ one-tail	1.71	

Table 3-6. Summary statistics for the canopy height models.

	Lidar	Nadir	Oblique	Nadir – lidar	Oblq. – lidar
Mean [m]	8.00	6.76	7.96	-1.24	-0.04
$\sigma$ [m]	3.85	4.61	4.27	3.02	2.74

Note: All units in this table are meters.

Table 3-7. Individual tree detection results from lidar CHM.

Spacing	Thinning	Genotype	Invntry	SWS	FWS	Detect	FP	FN	TP	r	p	F
B	NA	G1	57	5x5	7x7	48	1	10	47	0.82	0.98	0.9
B	NA	G2	58	5x5	7x7	58	0	0	58	1	1	1
B	NA	G3	59	5x5	7x7	56	1	4	55	0.93	0.98	0.96
B	NA	G4	59	5x5	7x7	57	0	2	57	0.97	1	0.98
N	0	G1	60	5x5	7x7	60	0	0	60	1	1	1
N	0	G2	65	5x5	7x7	63	0	2	63	0.97	1	0.98
N	0	G3	71	5x5	7x7	70	2	3	68	0.96	0.97	0.96
N	0	G4	69	5x5	7x7	67	1	3	66	0.96	0.99	0.97
N	0	G5	71	5x5	7x7	68	1	4	67	0.94	0.99	0.96
N	1	G1	140	5x5	7x7	137	6	9	131	0.94	0.96	0.95
N	1	G2	143	5x5	7x7	143	2	2	141	0.99	0.99	0.99
N	1	G3	141	5x5	7x7	135	1	7	134	0.95	0.99	0.97
N	1	G4	140	5x5	7x7	134	2	8	132	0.94	0.99	0.96
N	1	G5	140	5x5	7x7	131	1	10	130	0.93	0.99	0.96
W	0	G1	63	9x9	9x9	60	0	3	60	0.95	1	0.98
W	0	G2	71	9x9	9x9	71	0	0	71	1	1	1
W	0	G3	72	9x9	9x9	73	1	0	72	1	0.99	0.99
W	0	G4	69	9x9	9x9	68	0	1	68	0.99	1	0.99
W	0	G5	65	9x9	9x9	62	0	3	62	0.95	1	0.98
W	1	G1	109	9x9	9x9	108	0	1	108	0.99	1	1
W	1	G2	120	9x9	9x9	120	0	0	120	1	1	1
W	1	G3	121	9x9	9x9	121	0	0	121	1	1	1
W	1	G4	117	9x9	9x9	116	1	2	115	0.98	0.99	0.99
W	1	G5	119	9x9	9x9	117	0	2	117	0.98	1	0.99
Total			2199	-	-	2143	20	76	2123	0.96	0.99	0.98

Table 3-8. Individual tree detection results from nadir CHM.

Spacing	Thinning	Genotype	Invntry	SWS	FWS	Detect	FP	FN	TP	r	p	F
B	NA	G1	57	3x3	7x7	47	1	11	46	0.81	0.98	0.88
B	NA	G2	58	3x3	7x7	57	1	2	56	0.97	0.98	0.97
B	NA	G3	59	3x3	7x7	55	3	7	52	0.88	0.95	0.91
B	NA	G4	59	3x3	7x7	60	4	3	56	0.95	0.93	0.94
N	0	G1	60	3x3	7x7	58	0	2	58	0.97	1	0.98
N	0	G2	65	3x3	7x7	60	0	5	60	0.92	1	0.96
N	0	G3	71	3x3	7x7	72	6	5	66	0.93	0.92	0.92
N	0	G4	69	3x3	7x7	68	3	4	65	0.94	0.96	0.95
N	0	G5	71	3x3	7x7	72	3	2	69	0.97	0.96	0.97
N	1	G1	140	3x3	7x7	112	3	31	109	0.78	0.97	0.87
N	1	G2	143	3x3	7x7	135	2	10	133	0.93	0.99	0.96
N	1	G3	141	3x3	7x7	121	5	25	116	0.82	0.96	0.89
N	1	G4	140	3x3	7x7	135	2	7	133	0.95	0.99	0.97
N	1	G5	140	3x3	7x7	115	1	26	114	0.81	0.99	0.89
W	0	G1	63	7x7	7x7	60	0	3	60	0.95	1	0.98
W	0	G2	71	7x7	7x7	73	2	0	71	1	0.97	0.99
W	0	G3	72	7x7	7x7	82	10	0	72	1	0.88	0.94
W	0	G4	69	7x7	7x7	70	2	1	68	0.99	0.97	0.98
W	0	G5	65	7x7	7x7	61	1	5	60	0.92	0.98	0.95
W	1	G1	109	7x7	7x7	104	0	5	104	0.95	1	0.98
W	1	G2	120	7x7	7x7	119	0	1	119	0.99	1	1
W	1	G3	121	7x7	7x7	122	1	0	121	1	0.99	1
W	1	G4	117	7x7	7x7	117	2	2	115	0.98	0.98	0.98
W	1	G5	119	7x7	7x7	111	0	8	111	0.93	1	0.97
Total			2199	-	-	2086	52	165	2034	0.93	0.97	0.95

Table 3-9. Individual tree detection results from oblique CHM.

Spacing	Thinning	Genotype	Invntry	SWS	FWS	Detect	FP	FN	TP	r	p	F
B	NA	G1	57	5x5	7x7	48	3	12	45	0.79	0.94	0.86
B	NA	G2	58	5x5	7x7	58	4	4	54	0.93	0.93	0.93
B	NA	G3	59	5x5	7x7	63	7	3	56	0.95	0.89	0.92
B	NA	G4	59	5x5	7x7	56	3	6	53	0.9	0.95	0.92
N	0	G1	60	5x5	7x7	61	6	5	55	0.92	0.9	0.91
N	0	G2	65	5x5	7x7	67	9	7	58	0.89	0.87	0.88
N	0	G3	71	5x5	7x7	74	10	7	64	0.9	0.86	0.88
N	0	G4	69	5x5	7x7	64	3	8	61	0.88	0.95	0.92
N	0	G5	71	5x5	7x7	71	8	8	63	0.89	0.89	0.89
N	1	G1	140	5x5	7x7	134	18	24	116	0.83	0.87	0.85
N	1	G2	143	5x5	7x7	140	5	8	135	0.94	0.96	0.95
N	1	G3	141	5x5	7x7	134	16	23	118	0.84	0.88	0.86
N	1	G4	140	5x5	7x7	138	9	11	129	0.92	0.93	0.93
N	1	G5	140	5x5	7x7	130	9	19	121	0.86	0.93	0.9
W	0	G1	63	9x9	9x9	60	0	3	60	0.95	1	0.98
W	0	G2	71	9x9	9x9	71	0	0	71	1	0.97	0.99
W	0	G3	72	9x9	9x9	73	1	0	72	1	0.94	0.97
W	0	G4	69	9x9	9x9	68	1	2	67	0.99	0.94	0.96
W	0	G5	65	9x9	9x9	61	1	5	60	0.92	1	0.96
W	1	G1	109	9x9	9x9	102	0	7	102	0.94	1	0.97
W	1	G2	120	9x9	9x9	119	0	1	119	0.99	1	1
W	1	G3	121	9x9	9x9	126	6	1	120	1	0.9	0.95
W	1	G4	117	9x9	9x9	114	0	3	114	0.99	0.98	0.99
W	1	G5	119	9x9	9x9	115	0	4	115	0.94	0.99	0.97
Total			2199	-	-	2147	119	171	2028	0.92	0.94	0.93

## CHAPTER 4

### A STEM-BASED APPROACH TO SEMI-AUTOMATED ESTIMATION OF PINE MORPHOLOGY FROM LOW-ALTITUDE 3D MAPPING

#### **Introduction**

Forest management often uses remote sensing as a tool to provide accurate information on the morphology and spectral attributes of forests. An emerging platform for the remote sensing of forests is the unmanned aerial system (UAS), which consists of a remotely-piloted or preprogrammed robotic aircraft equipped with a remote sensing payload. UAS are quickly becoming the tool of choice for local-scale remote sensing projects due to their low entry cost and on-demand access to data collection, two traits that distinguish it from the more prevalent manned aircraft and satellite platforms. Particularly fascinating is the resurgence of photogrammetry that has accompanied the rise of UAS remote sensing. In parallel with the advent of accessible aerial platforms, the emergence of structure-from-motion (SfM) photogrammetry (Westoby et al., 2012) has made it possible for users to produce dense, three-dimensional (3D) reconstructions of the scene, similar to the data acquired via airborne laser scanning, using relatively inexpensive software to process photos taken with a commercial, off-the-shelf camera. The typically low flying heights of UAS missions allow for ground sample distances (GSDs) of the photos on the order of 1-3 cm, revealing the physical structure, or morphology, of the forest at a resolution that is unattainable from conventional remote sensing platforms. The problem now facing the photogrammetrist and forest manager is analyzing this rich source of 3D data.

#### **Aerial Photogrammetry in Forestry**

Forest inventory and mensuration, considering the area and density of the typical forest, is an arduous task. The “standard” for forest mensuration, the timber cruise, requires selective sampling by a forester in the field to measure and estimate forest parameters such as tree count,

diameter at breast height (DBH), tree height, stem form, and other physical traits (van Laar and Akça, 2007). Beginning in the 1920s, foresters tapped aerial photography as a mensuration tool, partnering with photogrammetrists to conduct “photo cruises” in the aim of producing stand maps (Spurr, 1948). Alongside these early methods, which relied greatly on interpretation by an expert of 2D photo mosaics, methods for interpreting stereopairs of images were also developed (Lyons, 1966), but never came to prominence as an effective forest sampling method. Despite the advances in photogrammetry since the middle of the 20th century, aerial photogrammetry conducted from a manned aircraft comes with inherent limitations. For reasons of cost effectiveness and safety, aerial photogrammetric missions are flown from considerable height above ground (typically >500 m), and the scale of the subsequent photographs limit the precision of subsequent analysis of the mission’s typical final product, the orthomosaic. For example, on a stand map created from conventional aerial photography, tree heights are determined manually through either photointerpretation or manual, 3D location of treetops via stereoscopic plotting. Without reliable information about the sub-canopy terrain, e.g. a digital terrain model, these tree heights are subject to interpretation—i.e., a good guess.

Developments in forest inventory and mensuration in the early part of this century included a fusion of airborne laser scanning (lidar) and low-altitude digital imagery (Bohlin et al., 2012; Korpela et al., 2007; St-Onge and Achaichia, 2001; Suárez et al., 2005). Both lidar and low-altitude aerial imagery allow for the 3D reconstruction of forests at the stand level (and, depending on the mission parameters, the individual tree level), which makes possible the recovery of allometric data through more indirect means than conventional, ground-based forest mensuration methods. Though these data can be acquired with the sensors on the ground (e.g. Kankare et al., 2013; Király and Brolly, 2007; Moskal and Zheng, 2011; Watt and Donoghue,

2005), the more efficient approach is to equip an aircraft with a sensor payload to cover a greater area from the aerial vantage. The spatial resolution of these data is dependent on, among other conditions, the distance between the sensor and the forest; however, the spatial resolution of the aerial data is often much greater than that available from satellite imagery.

The latest trend in forest inventory and mensuration is the use of automated systems, both for the collection and the processing of data (Dandois and Ellis, 2013; Karpina et al., 2016; Lisein et al., 2013; Tang and Shao, 2015; Watts et al., 2012; Yilmaz et al., 2016; Zarco-Tejada et al., 2014). Interest in the use of unmanned aircraft systems (UAS) has been growing in large part because they are more readily deployed and less cost prohibitive than manned aircraft. UAS can be equipped with a small sensor payload and can be flown over a sizeable area. Of particular interest is the ability of low-altitude UAS missions to collect data at a much higher spatial resolution than satellite-based and high-altitude, aerial-based sensors, which allows for the 3D reconstruction of forested areas in extraordinary detail.

### **Automated Forest Mensuration from Point Clouds**

Among the more intriguing developments in forest inventory and mensuration is the automatic determination of forest parameters from lidar or photogrammetric point clouds (Aschoff and Spiecker, 2004; Korpela et al., 2007; Maas et al., 2008; Simonse et al., 2003). The amount of data that can be acquired and analyzed is staggering when compared to what is possible via conventional means: UAS orthoimagery resolution is often on the order of 1-3 cm ground resolution, compared to ground resolution on the order of 1 m from high-resolution orthoimagery from conventional aircraft. Just as remarkable is the accuracy, repeatability, and efficiency with which these data can be collected. The process by which individual tree parameters such as stem count, DBH, and tree height (to name a few) can potentially be

identified is made possible not simply by advances in computing, but also by the increase in the amount of data regarding the 3D structure of the tree.

**Dense matching point clouds.** In earlier photogrammetric approaches, a handful of crucial 3D points (e.g. the highest point of a crown) were manually located by a trained operator using a stereoscopic plotter, an instrument for manually recording the 3D positions of objects in overlapping (or stereo) images. Area-based image matching methods, such as those used in auto aerotriangulation (AT), could automatically generate match points between consecutive stereo images, but only under certain geometric and radiometric conditions (e.g. non-convergent photography, presence of contrast/textture). The dense, accurate 3D reconstruction of the forested area via SfM—referred to herein as a “dense matching” point cloud—is more efficient than auto AT, in that it can produce feature matches in multiple, unordered images in various geometric configurations (Lingua et al., 2009). In a dense matching point cloud, that same tree is likely to have hundreds, possibly thousands, of 3D points associated with it; these results cannot be matched by conventional auto AT or a manual operator.

**UAS lidar point clouds.** Early airborne laser scanning (ALS) data exhibited low return density, which led to underestimation of tree heights (e.g. Nilsson, 1996). As lidar sensor technology advanced, the pulse rate and therefore return density attainable from lidar sensors increased to the point that sensing individual treetops became possible. The amount of subcanopy information attainable via ALS also increased as lidar sensor technology advanced to multiple-return and full waveform systems. ALS data progressed from offering a low-density picture of the top of forest canopy and the ground below to many data points’ (i.e., laser pulse returns’) worth of information per tree. Low-altitude UAS lidar data collection offers not only an

even-higher return density but also a higher probability of acquiring information about tree stems, a part of the tree that is often unattainable from high-altitude ALS data.

### **Canopy-Based Versus Stem-Based Individual Tree Detection**

From the richness of the low-altitude lidar or dense matching point cloud arises the challenge of interpreting the data. These low-altitude point clouds often contain information of the individual stems of trees, which is potentially valuable information for the forest manager or silviculturalist. Interpreting these stem data can be done manually, as certain computer programs allow a user to manually select points and make basic measurements within a point cloud. As presented above, some algorithms have been developed to interpret individual stems in a semi-automated manner in terrestrial lidar data. However, most methods of individual tree detection (ITD) from high-density point clouds are canopy-based. In many canopy-based methods, segmentation of individual trees in a stand is accomplished by applying image processing methods to a rasterized canopy height model (CHM) such as watershed segmentation (e.g. Wang et al., 2004) or local maxima detection (e.g. Mohan et al., 2017). One novel method exploits the subcanopy information present in high-density lidar data by applying a generalized normalized cut segmentation to voxelized (a sort of three-dimensional raster) point clouds (Reitberger et al., 2009). This study takes a novel approach, presenting algorithms designed to estimate a number of morphological traits of a forest stand—namely stem count, stem location, stem height, and DBH—by exploiting the presence of stem information in the point cloud.

## **Methods**

### **Data Collection and Processing**

**Baseline study site OSBS.** A baseline study was conducted at the Ordway Swisher Biological Station (OSBS) situated about 20 miles east of Gainesville, FL (Figure 4-1). A small, open portion of a slash pine (*Pinus elliottii*) plot with ten trees (average nearest neighbor spacing

of about 9 m) was selected for this pilot study. These trees range in size from 18 cm to nearly 40 cm DBH. A tripod-mounted Trimble GX terrestrial laser scanner (TLS) was used to scan the test plot from three scan stations. The GX's horizontal field of view is 360° and its vertical field of view is 60°, the latter of which was compensated for by scanning the study site from a greater distance to ensure full coverage of the trees. The angular resolution of each scan was set as to assure a minimum point spacing of 3 cm at the most distant tree in the scan; due to the varying range of trees from the scan stations and to overlap from the multiple scan stations, the average point spacing is > 3 cm. The data was processed and registered using Trimble PointScape.

The color (RGB) images for dense matching point cloud generation were acquired with a Nikon D60 DSLR with a focal length of 27 mm. Several photos were taken from a road adjacent to the study area in landscape orientation from approximately 1.7 m above the ground looking towards the targeted area. Figure 4-2 illustrates the locations and orientations of the exposure stations. Culling blurry and redundant imagery, a total of 179 photos were used to generate the photogrammetric point cloud using Agisoft PhotoScan. The photos were aligned using a self-calibrating bundle adjustment, and dense point cloud matching was carried out using “high” quality and “mild” depth-filtering settings. Both the TLS and dense matching point clouds were georeferenced to the local scanner coordinate system of the first TLS scan station.

This study is considered as a baseline because of the highly-controlled manner in which the data was collected, which ensured the resulting point clouds would be dense and exhibiting fewer errors. These data are used as a test site, so to speak, for developing and optimizing the detection and estimation algorithms presented below.

**Study site ACMF.** The next study was conducted at Austin Cary Memorial Forest (ACMF) near Gainesville, Florida, in the summer of 2016 (Figure 4-3). The study site within

ACMF is a roughly 2.2-acre stand of slash pine (*P. elliottii*) with a 2.7 m average nearest neighbor distance (i.e. stem spacing). Vertical RGB imagery was collected with the Canon EOS Rebel SL1 ( $f = 40$  mm) onboard a DJI S1000 vertical takeoff and landing (VTOL) UAS flying at 80 m above ground level (AGL). When the RGB imagery was initially collected, it was not expected that the resulting dense matching point cloud would feature stem information. This unexpected result was the catalyst for the study.

The RGB images were processed in Agisoft PhotoScan, and dense matching was performed at “medium” quality and “mild” depth-filtering settings. The point cloud was georeferenced with ground control points (GCPs) in the scene; the GCP locations were surveyed with an RTK GNSS system. The resulting dense, 3D point cloud was further processed using LASTools (Isenburg, 2012) to classify ground points and to calculate each point’s height above ground, effectively normalizing (or flattening) the terrain. From the point cloud of the entire study site, a 50×50 m subplot was selected for use in the algorithms described below. The subplot contains 69 tree stems (66 of which are alive) while containing approximately 70% fewer points than the point cloud of the entire study area and was deemed a suitable sample size for analyzing the algorithms’ performance while also reducing processing time.

**Study site Flatwoods.** The final study was conducted at Longleaf Flatwoods Preserve southeast of Gainesville, FL (Figure 4-4). The preserve features many stands of slash pine (*P. elliottii*) in various stages of growth. For this study, two circular plots 13 m in radius were selected. The two plots are thinned plots of 28 and 29 stems, respectively, with an average nearest neighbor distance of about 3 m. The stems average about 18 cm DBH and 17 m in height. The row-planted stand has 12 ft (3.6 m) row spacing where every fourth row has been thinned.

Lidar data was collected over the site using the GatorEye Unmanned Flying Laboratory, a sensor suite aboard a DJI Matrice 600 VTOL UAS. The lidar sensor is a Velodyne VLP-16 PUCK Lite, a dual-return system with a 300,000 points/s pulse rate (Figure 4-5). The lidar data was collected from a height of 40 m AGL. The point cloud was directly georeferenced by a process similar to that outlined in Chapter 1. The resulting point cloud has an average point density of 450 points/m<sup>2</sup>. This point cloud was also normalized for subsequent analysis using the method described above.

**Timber cruise data.** To serve as ground truth for the subsequent algorithm estimations, a timber cruise was conducted of both main study sites. The DBH of each stem was determined with a tree tape and stem heights determined with a hypsometer. (Tree heights were not recorded at the OSBS site, as those data were used only for the development of the stem detection and DBH estimation algorithms.) For the ACMF site, stem locations were determined via horizontal angle offsets with a total station. Stem locations at the Flatwoods site were initially found using a compass and a sonar-based hypsometer from the plot centers, but these measurements suffered from systematic biases in both range and bearing and were replaced by stem locations taken directly from the lidar point cloud.

### **Stem Detection Algorithm**

A natural thought is to cluster the points according to their  $(x, y)$  coordinates. The algorithm developed by Maas et al. (2008) is adopted. The main idea of this method is to cluster points using a moving window in the  $(x, y)$  plane (Figure 4-6). First, all the points within a slice are projected onto the  $(x, y)$  plane. The square window, which is divided into quadrants, begins in an arbitrary position within the slice. If the number of points within the window is greater than a predetermined threshold, the algorithm deems those points as belonging to a cluster. The

window moves by one-half its length in the plane towards the quadrants which contained data points, and the algorithm repeats, terminating once it reaches an area where the moving window contains fewer than the threshold number of points. Clusters that are adjacent to one another are determined to be the same cluster. A circle is then fitted to each of the final clusters; if the fitted circle is above a minimum threshold radius (to avoid detection of smaller objects), and the root mean-square error (RMSE) of the fitted circle is below a predetermined maximum, the cluster is determined to be from a tree stem.

Although the circle fitting test could identify some of the stems, clusters which fail the test also have the possibility of containing stems. For instance, when a stem is tightly surrounded by high understory growth, it is probable to cluster the stem with the understory growth, which causes the cluster to fail the fitting test. Thus, the stem detection method presented here augments the moving window strategy presented by Mass et al. by performing a Hough transform on those clusters which were initially not identified as stems (Figure 4-7). The Hough transform is a robust method for detecting a model—in this case, a circle—in point data.

For both the Maas et al. algorithm and the Hough transform, the centers of the detected circles are recorded. Those detected circles whose centers are within 0.5 m of each other are deemed to be from the same tree stem, and the points from each of the grouped, detected circles are combined to form the detected stem. A flowchart of the entire stem detection algorithm is shown in Figure 4-8.

### **Diameter at Breast Height Estimation Algorithm**

The detected stems are passed to a RANSAC (Fischler and Bolles, 1981) cylinder fitting algorithm for the sake of finding the stem's diameter at breast height (DBH). The algorithm is based on the cylinder fitting method presented by Schnabel et al. (2007). In the Schanbel et al. method, two points and their associated surface normals are chosen at random. The two lines

defined by each point and its corresponding surface normal are projected onto the plane defined by the cross product of the two normals. The intersection of the two lines is taken to be a point along the axis of the cylinder, and the direction of the cross product is the direction of the axis (Figure 4-9). The distance from the first of the randomly-chosen points to the axis is taken as the radius of the cylinder. This process is repeated  $t$  times, where

$$t = \text{ceil} \left( \frac{\ln(1-p)}{\ln(1-(1-r)^n)} \right) \quad (4-1)$$

In the above equation,  $r$  is the predefined percentage of points expected *a priori* to be outliers [0,1). The value  $n$  is the number of points that selected to fit to the model. (In this case, the model is a cylinder, and the number of points randomly selected is 2.) The value of  $p$  is the certainty [0,1) that, once all the trials have run, that at least one trial did not contain any outliers. For this study,  $p = 0.999$ .

For each iteration, the RMSE of the residuals of all points to the RANSAC estimated cylinder is calculated. After all iterations, the initial cylinder with the lowest RMSE is selected, and the  $(1 - r)$  percentage of points with the lowest residuals are fitted to a final cylinder via ordinary least squares (OLS). As a final step, only those points within the predefined height threshold above and below breast height are passed to this algorithm. This threshold typically does not extend too far below breast height as to avoid understory growth; however, the threshold above breast height can be adjusted more freely to ensure sufficient points are passed into the cylinder fitting algorithm.

The RANSAC cylinder fitting algorithm for this study has been augmented from the Schnabel et al. method in three ways:

1. When two random points (and their associated normals) are selected for the initial cylinder fitting, the points are not kept unless the angle between the normals is between  $30^\circ$  and  $150^\circ$ ;

2. the initial cylinder fit is not kept unless the cylinder is oriented near-vertically (i.e., the unit vector representing the direction of the axis must have a z-component  $> |0.9|$ ); and
3. as described above, an OLS-fitted cylinder is applied to the inlier points as identified in the RANSAC process. Double the radius of the final OLS-fitted cylinder is taken as the estimate for DBH.

### **Height Estimation Algorithm**

The locations of the detected stems are used as an initial search area for the local maxima of the entire dense point cloud. Using LASTools, all points whose ( $x, y$ ) coordinates (i.e., horizontal positions) are within 0.5 m of the stem's horizontal location are kept, and the point with the highest height above ground is taken as the highest point of the tree stem. This method is not a general solution for estimating the heights of stems. For example, an error in the stem's horizontal position, or a significant lean of the stem, could cause the horizontal position of the detected local maximum to not fall within an 0.5-m radius of the horizontal position of the stem. Assuming the stem locations are accurate, the proposed method for estimating the heights of stems will be sufficient, as no stems in the study area exhibit significant lean. For two stems whose tops are within 0.5 m of each other, this algorithm would record only the height of the taller stem. For two stems to be this close together is an unlikely scenario based on *a priori* information about the pine stands used for this study.

### **Equivalence Testing**

To determine if the DBH and height estimation algorithms are effective at estimating the average DBH and stem height at the stand level, the means of the cruise and estimation data will be subject to equivalence testing, specifically, the two one-sided t-test (TOST). This test begins with the null hypothesis that the means are not “equivalent,” (i.e. not significantly different) with the alternative hypothesis being that the means are “equivalent” within some predetermined, “tolerable” limit  $\delta$ . Put briefly, TOST finds the confidence interval for the difference between

the two means; if that confidence interval lies entirely inside the interval  $[-\delta, \delta]$ , then the two means are said to be “equivalent” (Limentali et al., 2005). The confidence interval for the difference between two means is found by:

$$(\bar{x}_1 - \bar{x}_2) \pm t_{(1-\frac{\alpha}{2}, n_1+n_2-2)} \sqrt{s_p^2 \left( \frac{1}{n_1} + \frac{1}{n_2} \right)} \quad (4-2)$$

where

$$s_p^2 = \frac{(n_1-1)s_1^2 + (n_2-1)s_2^2}{n_1+n_2-2} \quad (4-3)$$

For this study, the limit of  $\delta = 0.1 \times \bar{x}_{cruise}$  will be used to determine if the means of the DBH and height estimation results are within 10% of the cruise data.

## Results

### **Terrestrial Data, Preliminary Study**

As the stem detection algorithm was adapted from a method developed specifically for TLS data, results were expectedly positive. The algorithm detected all ten of the stems in both the lidar and dense matching clouds under various configurations of slice height, slice minimum, and the tunable parameters of the algorithm (i.e., search window size, radius threshold, goodness of fit). These preliminary tests provided useful guidance for efficient use of the algorithm:

1. Excluding near-ground and crown data slices from the stem detection algorithm led to runtime improvements of over 95%. The majority of data in those slices are non-stem data which unnecessarily slows the clustering subroutine. Adding a low and high height threshold provides a significant advantage.
2. The speed of the cluster detection routine is inversely proportional to the size of the search window. Increasing the size of the search window from the Mass et al. recommendation of 20 cm to 50 cm led to a further 10% improvement in runtime.

As mentioned previously, the stem detection algorithm will return only those points deemed to be part of the stem, which is not necessarily a depiction of the entire stem (Figure 4-10). Portions of the stem below or above the height thresholds set in the pre-processing stage, as well as

portions within the thresholds that were not deemed circular during the clustering, will not be returned as part of the stem.

For the preliminary testing of the DBH estimation algorithm, the aim was to find the ideal outlier ratio. Numerous trials were performed on both the lidar and dense matching detected stems at outlier ratios ranging from 70% to 1%, with those trials with an outlier ratio 40% and lower showing favorable results (Figure 4-11).

Although these trials proved useful for developing the algorithms, they did not particularly challenge the robustness of the algorithms against noisy data. From the terrestrial pose, after reasonable outlier filtering was performed, both the TLS and terrestrial dense matching point clouds exhibited little noise.

### **Dense Matching, ACMF Site**

Stem detection performed reasonably well within the UAS dense matching cloud at the ACMF site (Figure 4-12). With a slice height of 20 cm and a slice minimum of 5, the algorithm detected 61 of the 69 stems in the study plot. Of the 61 detected stems, 58 were alive and included in the timber cruise.

The DBH estimation for these 58 stems proved to be unreliable at the individual tree level. A range of outlier ratios from 70%-1% were attempted, with 10% producing the least inaccurate results. Linear regression with a fixed intercept comparing the cruise and estimated DBH, surprisingly, returned a negative value of  $r^2$  (-0.34) (Figure 4-13). Cylinder fitting failed on five of the 58 stems. Comparison of the means was somewhat more promising (cruise 20.8 cm,  $\sigma = 5.1$  versus detected 22.9 cm,  $\sigma = 12.9$ ), but in the TOST, the null hypothesis of inequivalence was not rejected at  $\alpha = 95\%$  ( $p$ -value 0.51).

Estimating the heights of the trees in the study plot showed a somewhat greater level of success, presenting an  $r^2$  of 0.44 in the fixed intercept model (Figure 4-14). The means also fail to reject the null hypothesis of the TOST at 95% ( $p = 0.07$ ). To what extent this can be attributed to a possible underestimation of the heights as measured by hypsometer cannot be said (cruise mean 18.8 m versus detection mean 19.8 m), but the regression fit does strongly suggest the inadequacy of the dense matching cloud to provide reliable stem heights on an individual tree level.

### **Lidar, Flatwoods Site**

Stem detection proved more difficult with the lidar point cloud (Figure 4-15). Despite the ability of lidar to penetrate canopy, the stems in the scene tended to consist of fewer points than those in the dense matching cloud from the ACMF site. The optimal configuration for stem detection for the Flatwoods plots proved to be 200 cm slices, with a slice minimum of 1. Regardless, only 40 of the 56 stems were positively detected.

The DBH estimation algorithm performed similarly in the lidar data as it did in the dense matching data, showing no reliability at the individual level ( $r^2 = 0.03$ , fixed intercept) (Figure 4-16) and overestimating at the plot level (20.1 cm,  $\sigma = 3.6$  to 24.0 cm,  $\sigma = 10.5$ ). DBH estimation fails to reject the null hypothesis of inequivalence in the TOST at 95% ( $p = 0.77$ ). For both the dense matching and lidar data, the variance of the DBH estimations is much higher than the variance of the cruise data.

Height estimation also performed similarly in the lidar and dense matching data, exhibiting unreliable precision for estimation of the heights of individual stems ( $r^2 = -0.22$ ) (Figure 4-17) while overestimating heights at the stand level (18.9 m mean detection versus 17.7 m mean cruise). The means in the TOST do not reject the null hypothesis of inequivalence at

95% ( $p = 0.13$ ). For both the dense matching and lidar height estimations, the variance of the estimations is lower than that of the cruise data. Summaries of stem detection, DBH estimation, and height estimation for both the ACMF and Flatwoods sites can be found in Tables 4-1:3.

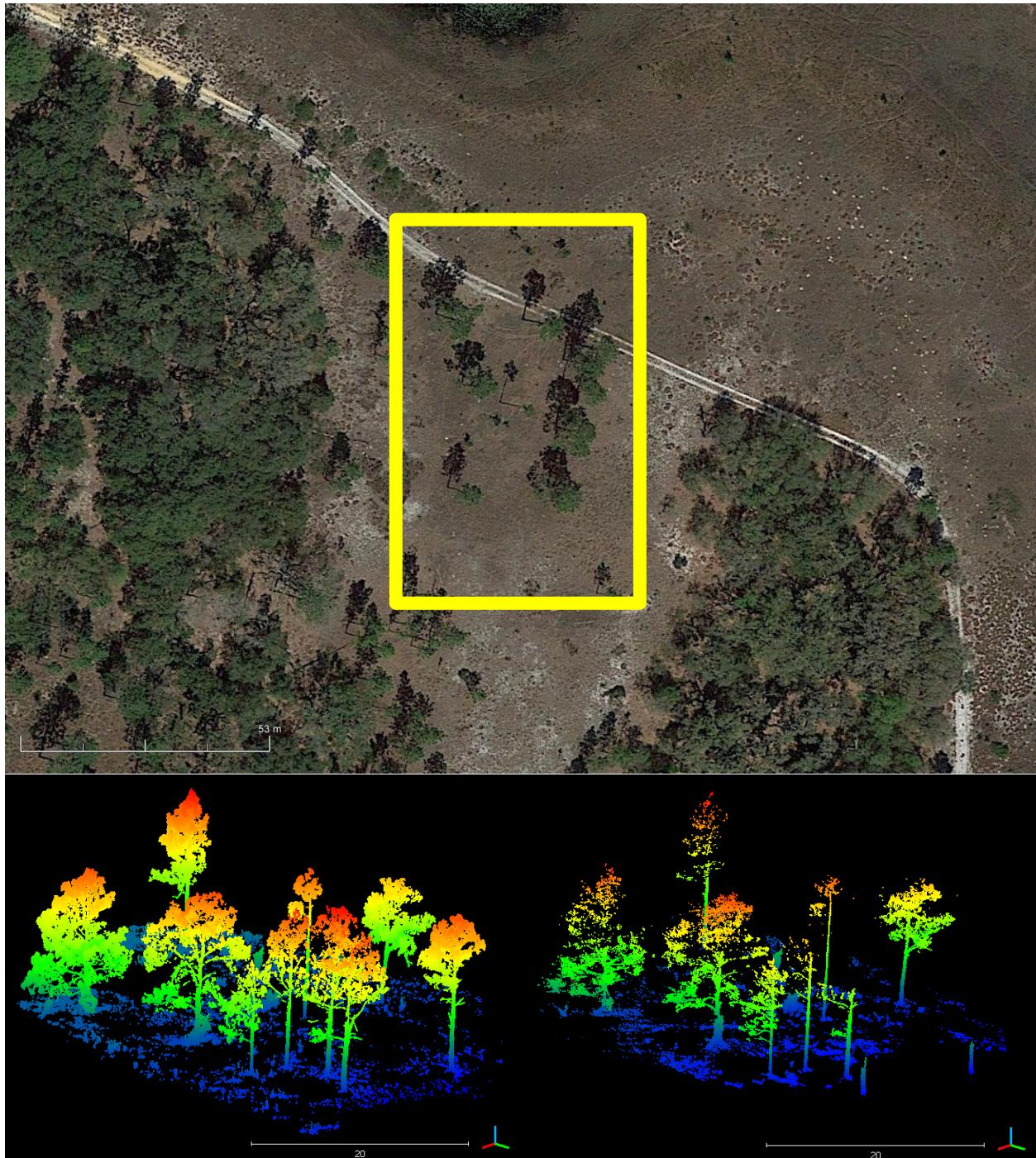


Figure 4-1. Baseline study site at Ordway Swisher Biological Station near Gainesville, FL. TOP: Satellite view; the stems used are outlined in yellow. Image courtesy of Google. BOTTOM: Oblique views of the TLS (left) and dense matching (right) point clouds.

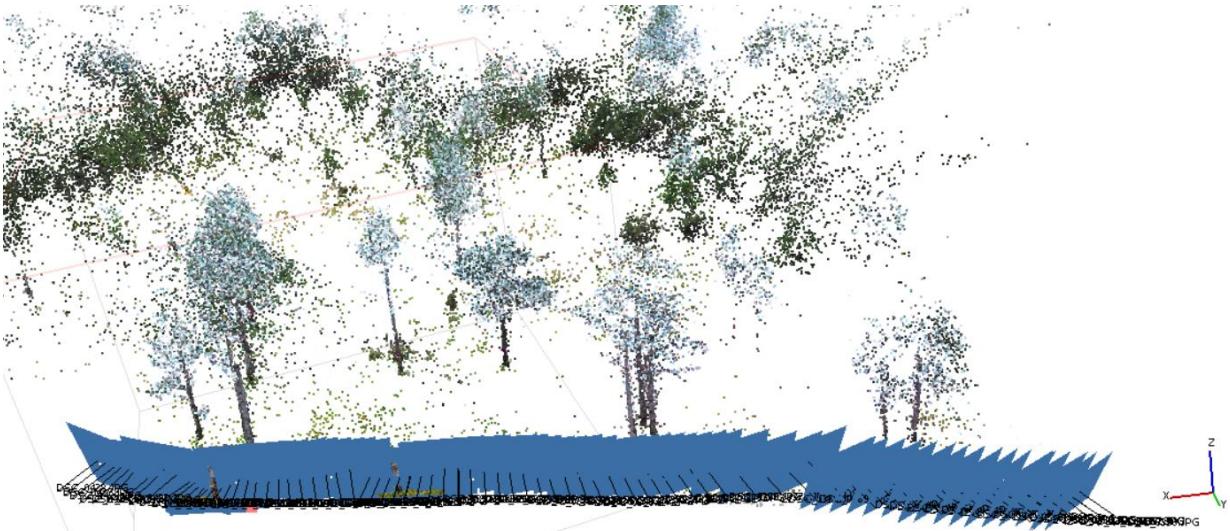


Figure 4-2. Locations and orientations of exposure stations for terrestrial dense matching point cloud.



Figure 4-3. Oblique view of ACMF dense matching point cloud. One of the obstacles that the stem detection and cylinder fitting algorithms must overcome is missing information along the tree stem, which is due mainly from occlusion from the canopy. Though the point cloud retains color information from the RGB photos, these data are not used in the algorithms.

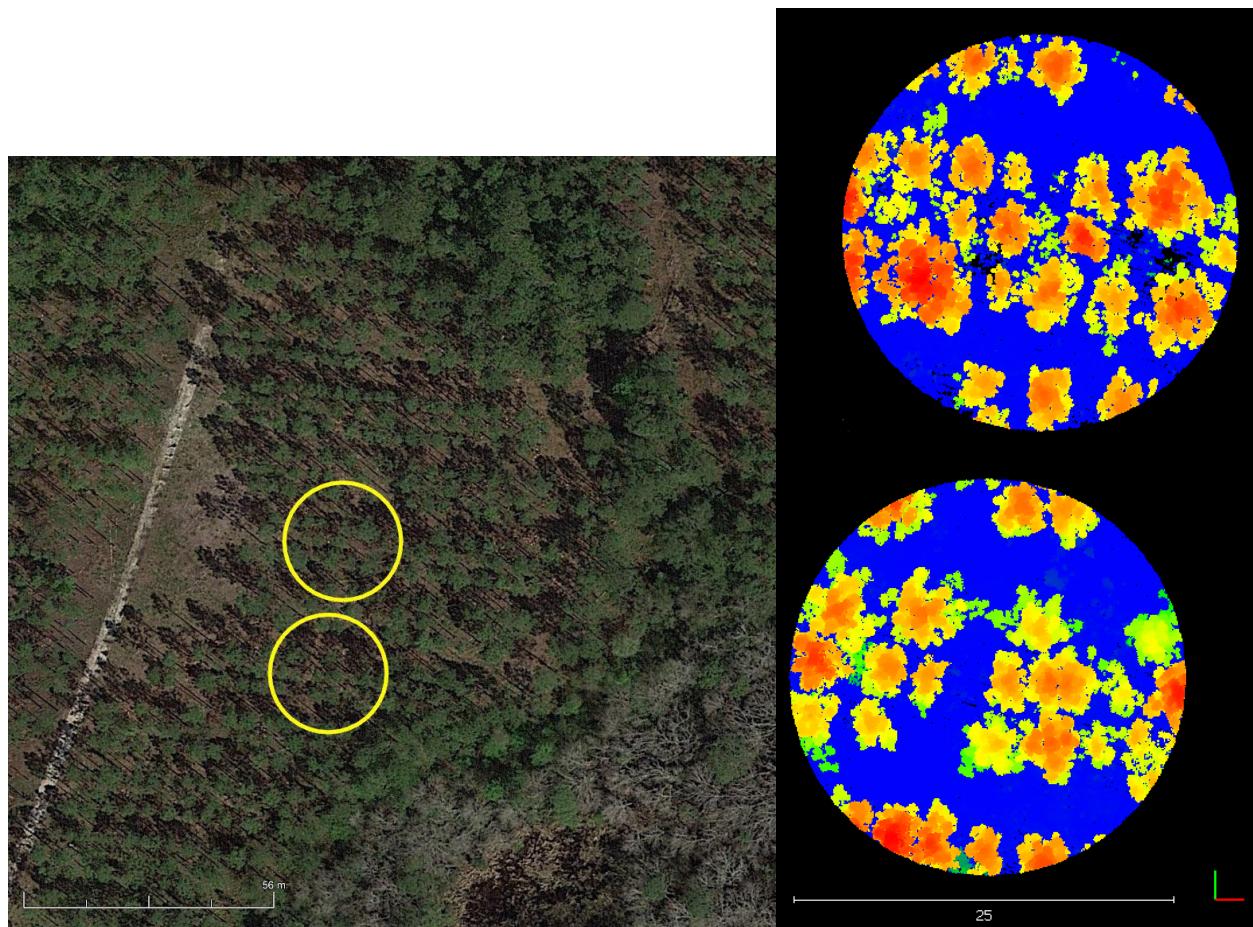


Figure 4-4. Longleaf Flatwoods Reserve study site. LEFT: Satellite view with study plots outlined in yellow. Image courtesy of Google. RIGHT: Plan view of lidar point cloud colorized by height.



Figure 4-5. GatorEye Unmanned Flying Laboratory (UFL). The GatorEye UFL is a sensor suite shown here on board the DJI Matrice 600. Image courtesy of Dr. Eben Broadbent ([www.spaclab.org/gatoreye.html](http://www.spaclab.org/gatoreye.html)).

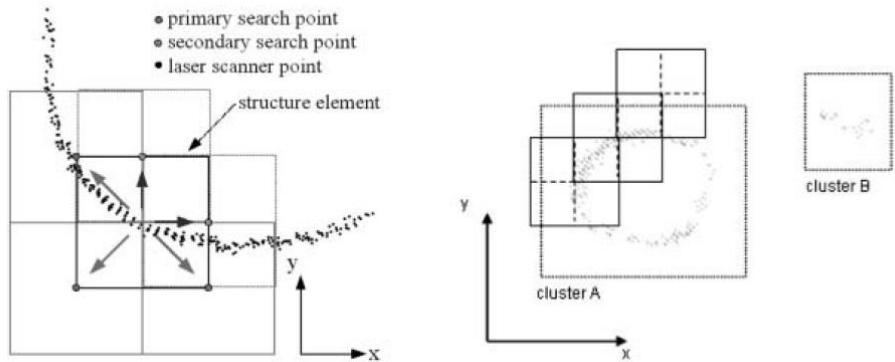


Figure 4-6. Moving window search strategy. The points in the point cloud are here labeled as “laser scanner points,” though in the proposed study, the data points are from the 3D reconstruction of the scene through SfM photogrammetry (Maas et al., 2008).

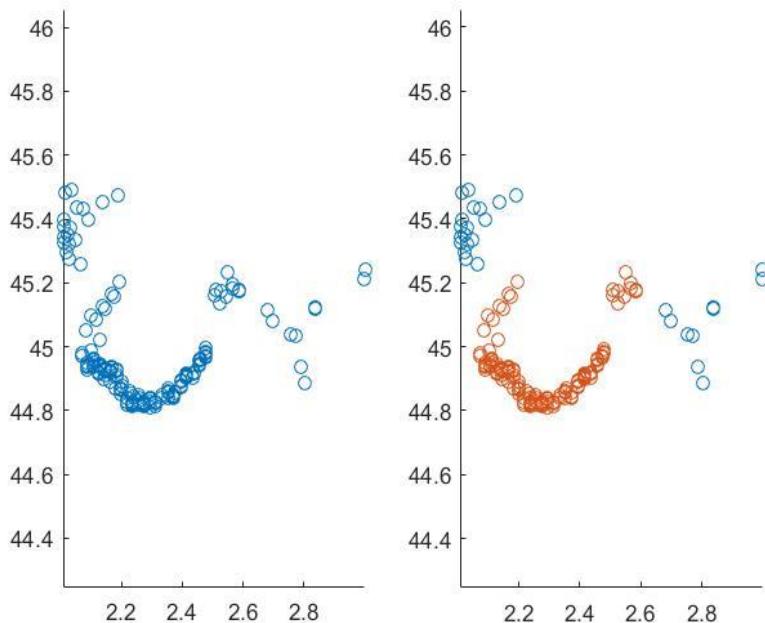


Figure 4-7. Circle recognition by Hough transform. Depicted is a plan view of a cluster not deemed a tree stem by the Maas et al. method. The points are shown here in a planimetric (top-down) view, with the points determined by the Hough transform to be from the tree stem shown in orange (right).

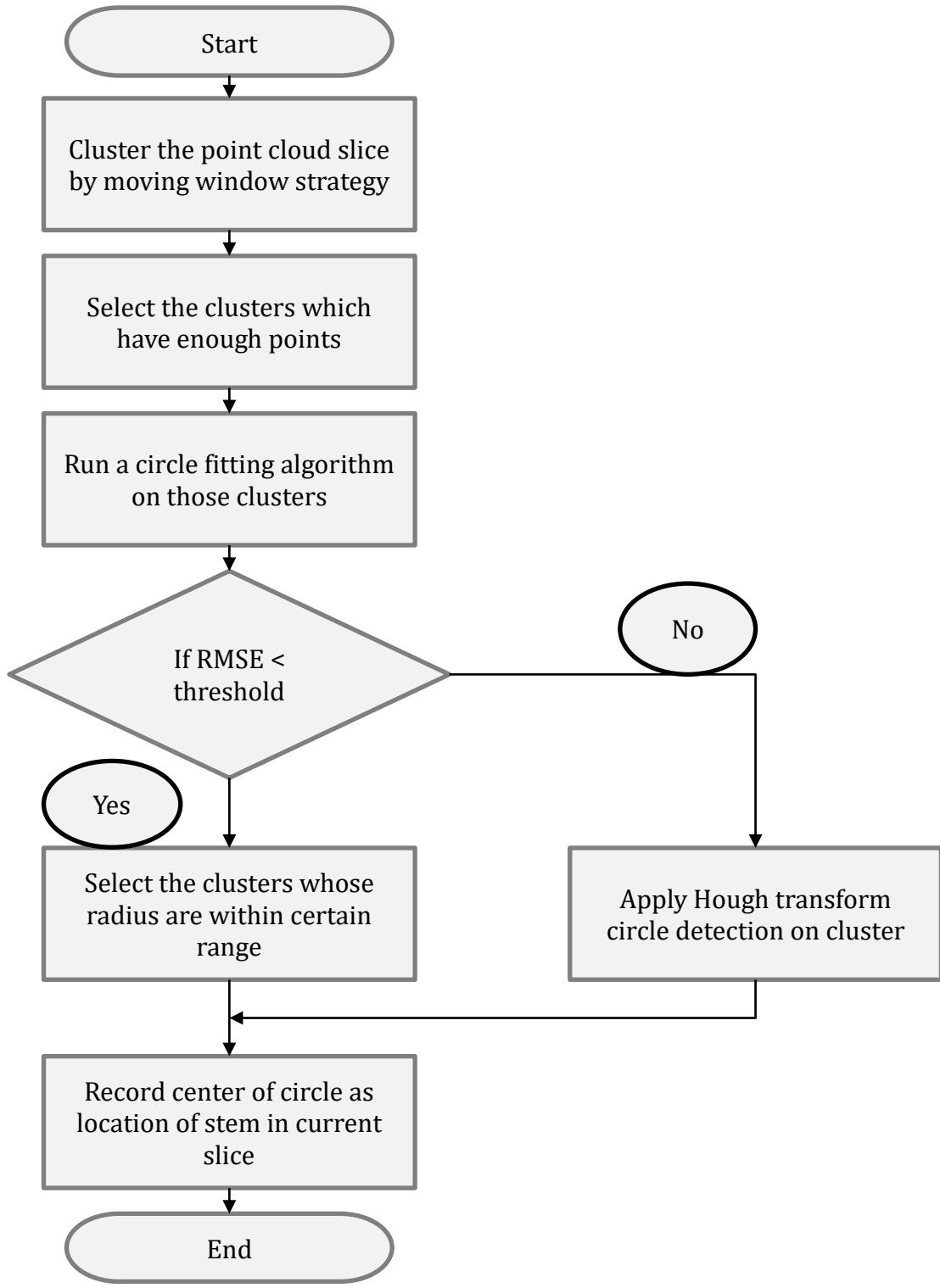


Figure 4-8. Flowchart of the stem detection method. Once this algorithm has been repeated at all horizontal slices, the recorded centers undergo a final clustering to determine which detected circles belong to the same tree stems.

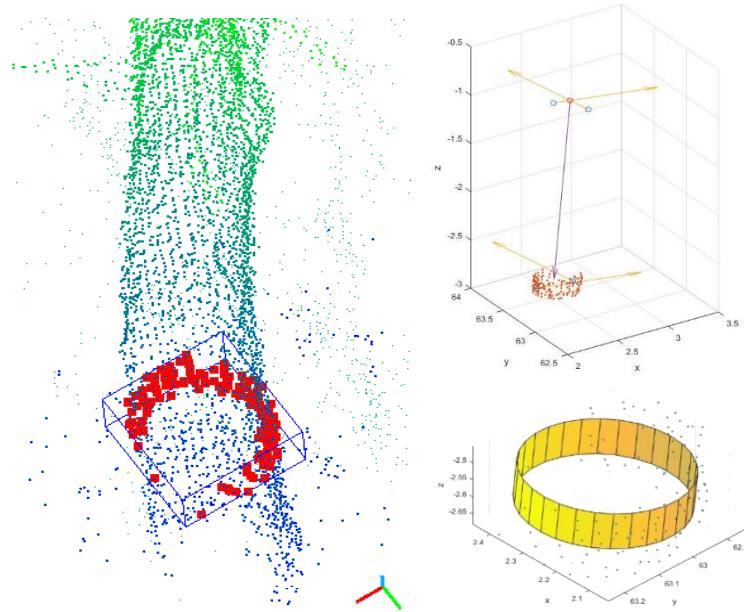


Figure 4-9. RANSAC cylinder fitting. A selection of points from the detected stem around breast height (1.4 m) is selected (shown in red, left); these points are passed to the cylinder fitting algorithm, where the initial RANSAC cylinders (top right) and final least-squares cylinder (bottom right) are found.

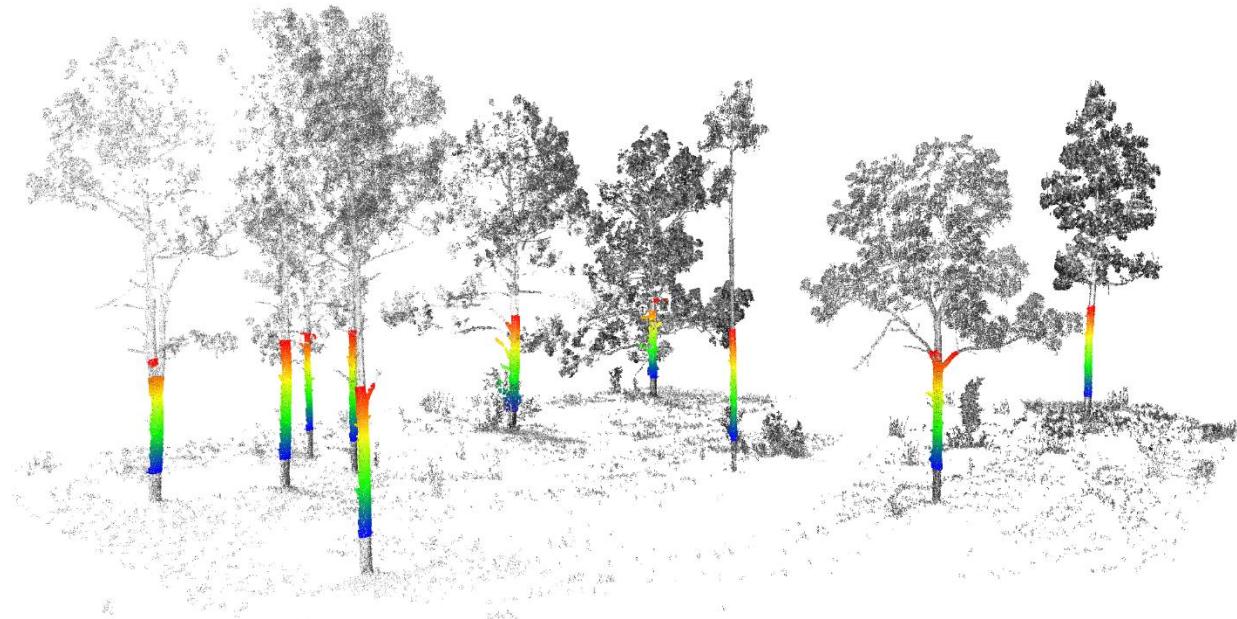


Figure 4-10. Stem detection results, TLS data, OSBS study site.

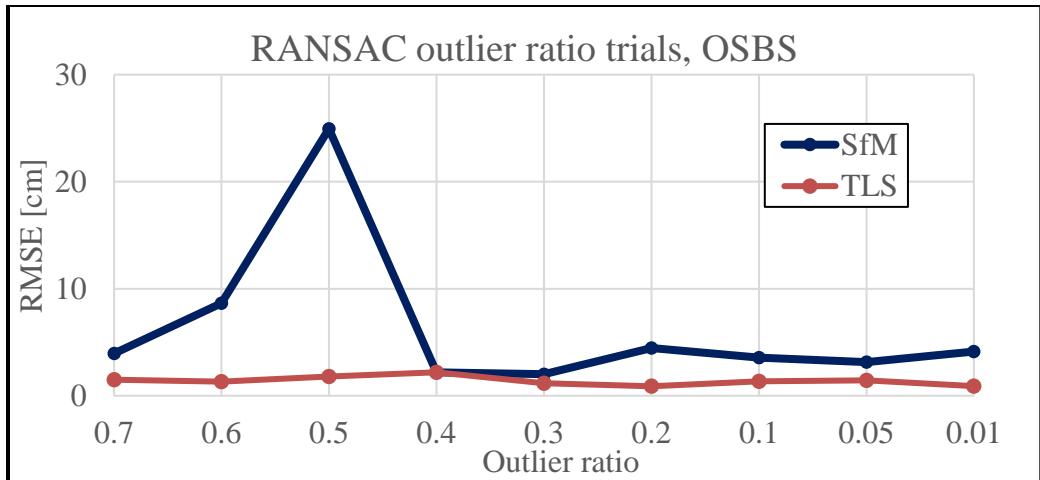


Figure 4-11. Outlier ratio trials for RANSAC DBH cylinder fitting for DBH estimation.

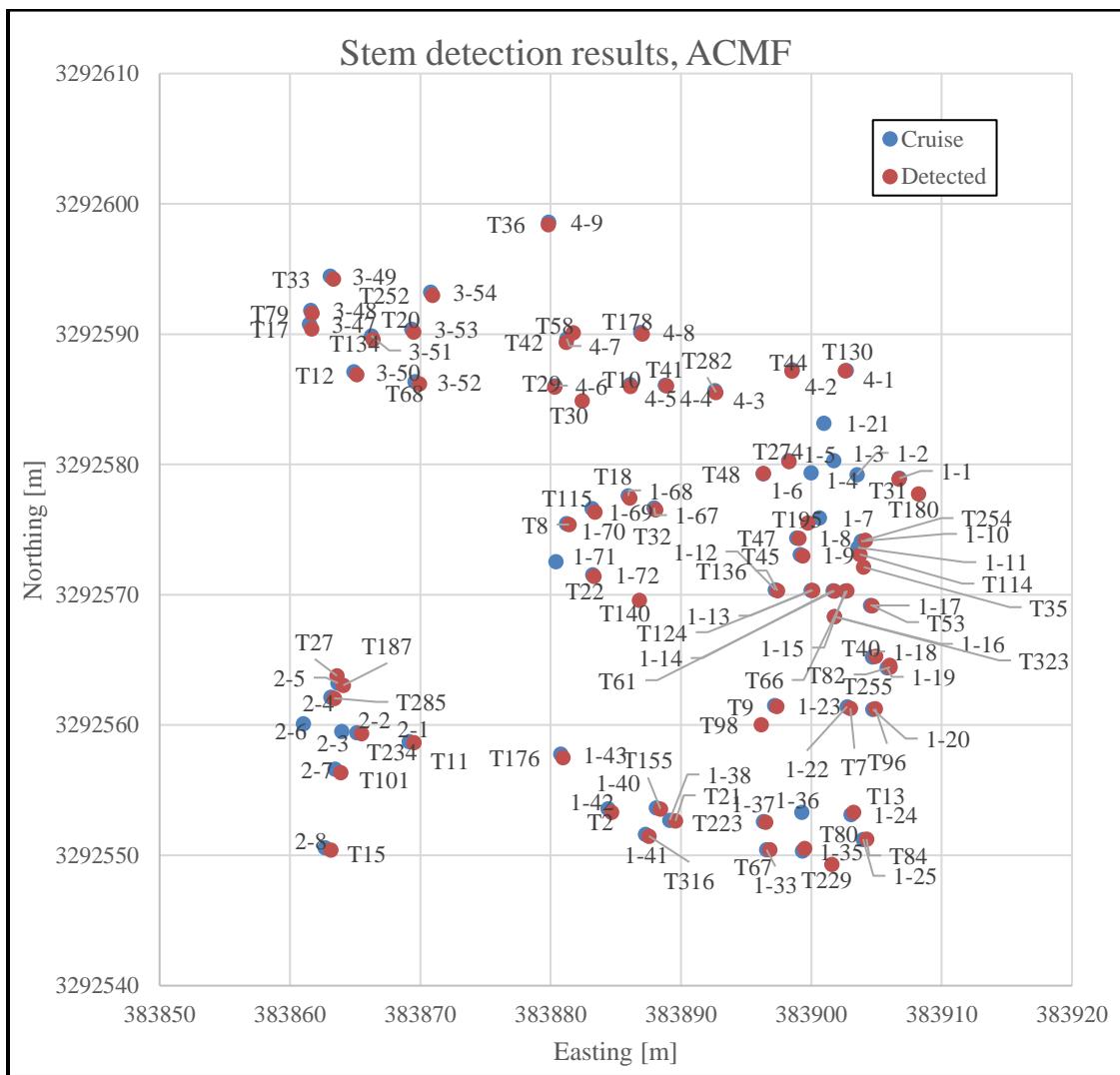


Figure 4-12. Stem detection results, dense matching cloud, ACMF study plot.

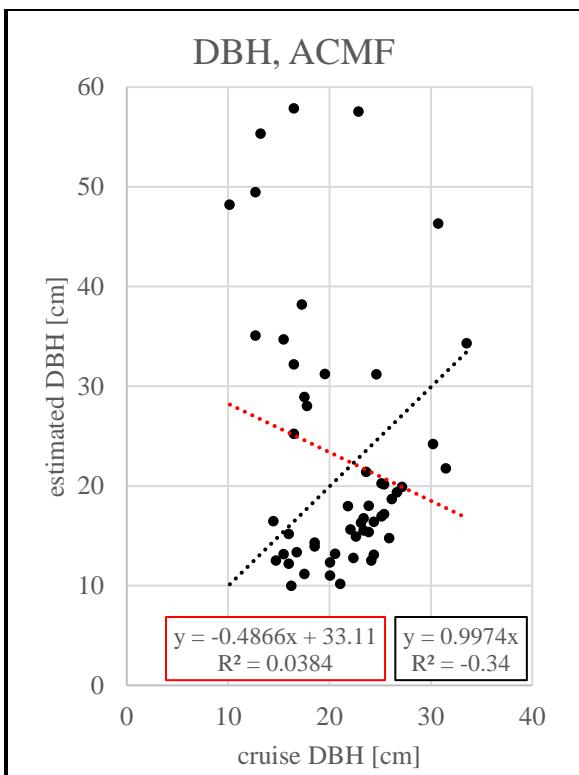


Figure 4-13. DBH estimation results, dense matching cloud, ACMF study plot. Fixed intercept linear regression line in black; linear regression without fixed intercept in red.

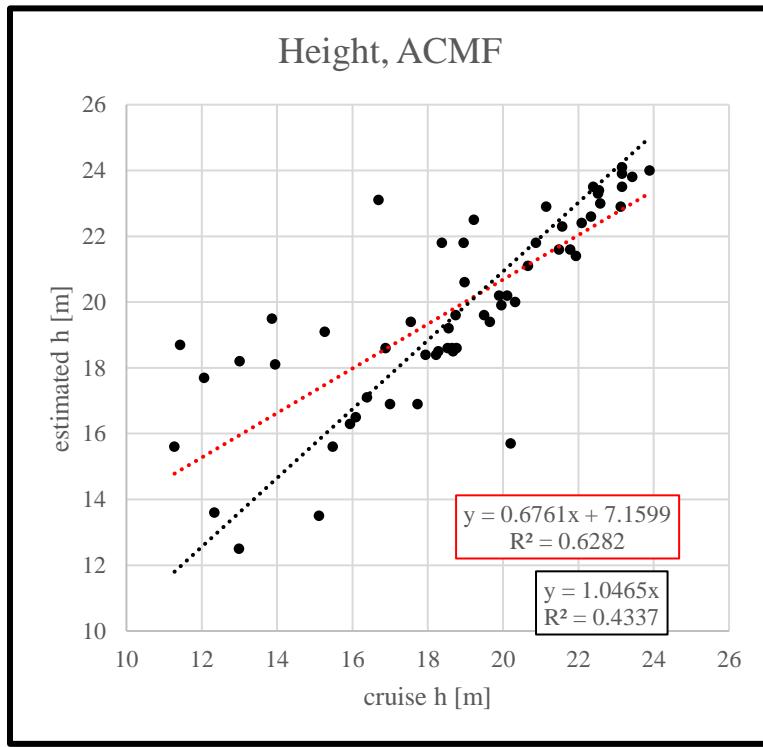
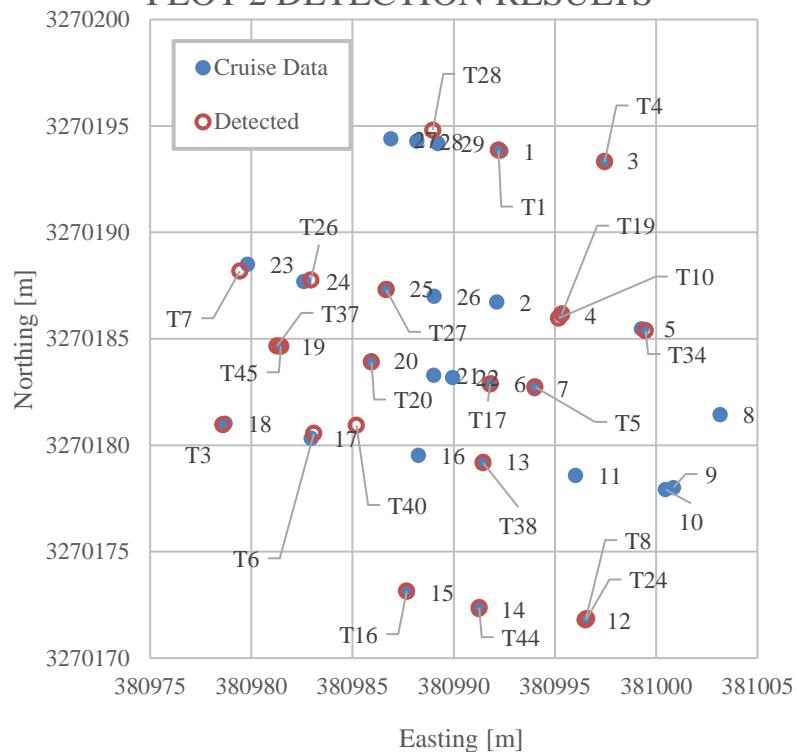


Figure 4-14. Height estimation results, dense matching cloud, ACMF study plot. Fixed intercept linear regression line in black; linear regression without fixed intercept in red.

### PLOT 2 DETECTION RESULTS



### PLOT 3 DETECTION RESULTS

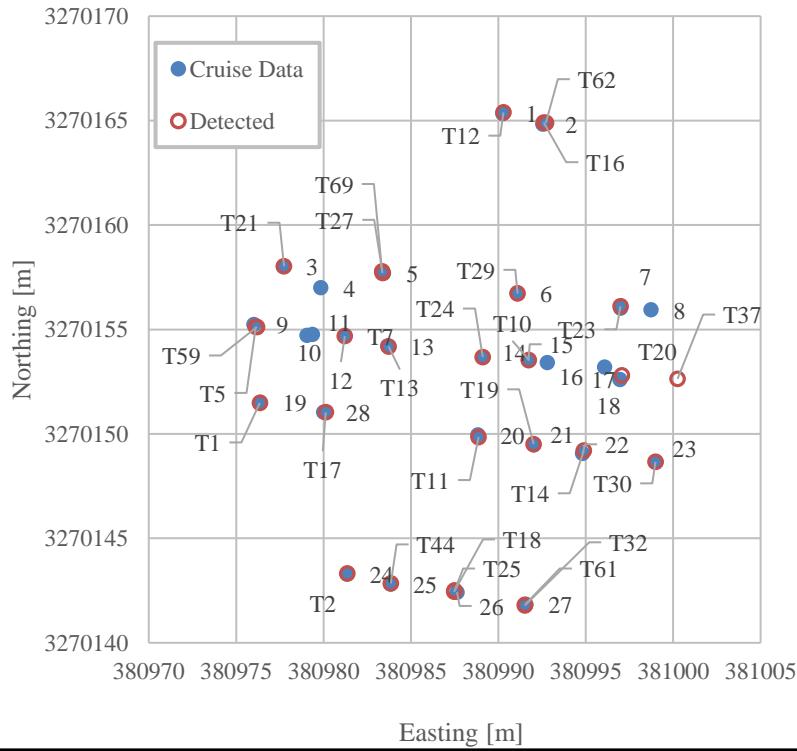


Figure 4-15. Stem detection results, lidar cloud, Flatwoods site.

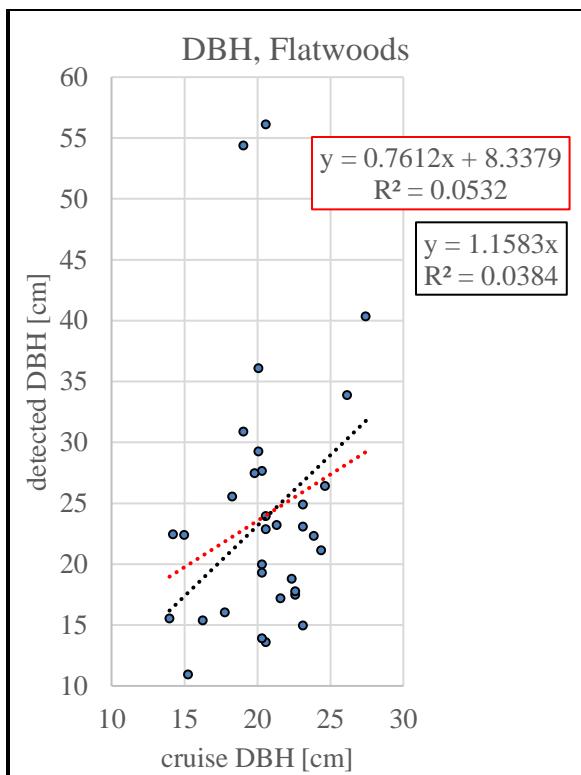


Figure 4-16. DBH estimation results, lidar cloud, Flatwoods study plots. Fixed intercept linear regression line in black; linear regression without fixed intercept in red.

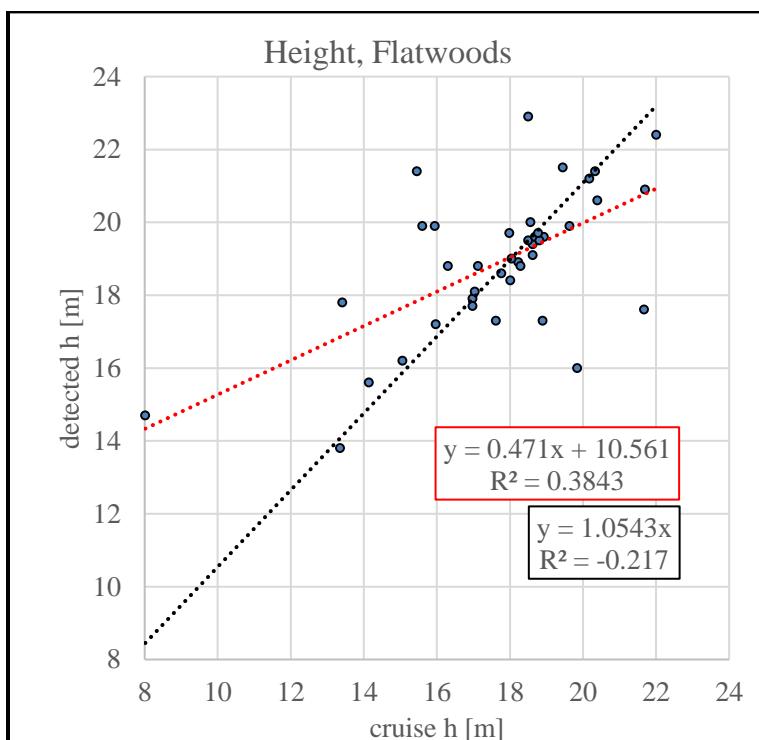


Figure 4-17. Height estimation results, lidar cloud, Flatwoods study plots. Fixed intercept linear regression line in black; linear regression without fixed intercept in red.

Table 4-1. Stem detection summary.

Match type	Dense matching, ACMF	Lidar, Flatwoods
True positive	58 (61)	40
False positive	8 (5)	10
False negative	8	18

Note: Three of the stems detected at the ACMF site were not included in the cruise because they were dead. They are included here because the stem detection algorithm successfully detected their presence in the point cloud.

Table 4-2. DBH estimation summary.

	Dense matching, ACMF		Lidar, Flatwoods	
	DBH_c	DBH_e	DBH_c	DBH_e
Mean	20.78	22.87	20.10	23.95
$\sigma$	5.10	12.93	3.56	10.54

Note: All values in cm. “c” denotes cruise values and “e” denotes estimated values.

Table 4-3. Height estimation summary.

	Dense matching, ACMF		Lidar, Flatwoods	
	h_c	h_e	h_c	h_e
Mean	18.74	19.83	17.74	18.92
$\sigma$	3.37	2.87	2.60	1.97

Note: All values in m. “c” denotes cruise values and “e” denotes estimated values.

## CHAPTER 5 CONCLUSION

### **Recommendations for UAS 3D Remote Sensing**

In Chapter 2, the importance of “minding the gaps” is the primary focus of the study. These gaps are most likely to occur as overlap decreases (or overlap is not present), forward velocity increases, and/or the ground is nearly flat. Overlap obviously helps fill in any potential gaps. As forward velocity decreases not only does point density increase, but random, back-and-forth yaw/crab of the airframe helps to fill in gaps. Varying topographic relief tends to break up the repeating gap patterns as well, though as is also the case in conventional ALS, drastic drop-offs such as cliffs or ravines, if not accounted for, can lead to poor coverage of the lower terrain. The gaps were shown to have more potential to provide degrading quality of coverage as the rotation rate of the scanner head decreased. Velodyne allows the user to adjust the rotation rate between 5-20 Hz. A lower rotation rate may lead to decreased power consumption (thus longer flight times), though this was not confirmed as a part of this study. Concerns of gaps in coverage, especially with respect to flying height, forward velocity, and the size of objects of interest in the scene, should be considered in alongside the potential benefits of decreasing the rotation rate of the scanner head.

In Chapter 2 it is also suggested that the maximum range of returns be set at some limit below the VLP-16’s empirical maximum range of 120 m. In practice, this has been found to produce vastly more accurate results. However, this finding has been confined to primarily research missions with narrow flight line spacing and slow forward velocities, where maximizing area of coverage was not of concern. This should be weighed carefully when planning a UAS lidar mission.

In chapter 3, Agisoft PhotoScan was used for processing both nadir and oblique imagery. However, it is not immediately clear what method PhotoScan uses for initial approximations of camera orientation without *a priori* information from direct georeferencing; the manual suggests that nadir (i.e. vertical) images are preferred, or at least expected by default. This notion is supported by the initial difficulties encountered in processing the oblique images. Using the angular attitude data for the oblique images as initial approximations during the image alignment (i.e. bundle adjustment) step of the workflow in Agisoft PhotoScan resulted in more photos being aligned. Image alignment without these initial approximations resulted in roughly two dozen photos being rejected out of about 500. However, the rotation data from the Xsens Mti-G-710 as reported was not compatible with the rotation data accepted by PhotoScan. Conversion of the Xsens rotation data from yaw-pitch-roll in the Easting-Northing-Up (ENU) frame to yaw-pitch-roll in the Northing-Easting-Down (NED) frame was not trivial; the procedure is detailed in Appendix E.

It is important to note for both lidar and photogrammetric UAS data collection, the mission planning should take into account the average and maximum heights of the targets of interest in the scene. Overlap, side lap, point density, etc. can be drastically different at the ground and target ranges from sensor. For example, for a UAS remote sensing mission flown 50 m AGL, with trees in the scene averaging 15 m, considerations for point coverage, overlap, etc. should be made for both 50 m and 35 m AGL. Especially in a forested scene, critical information could be lost if image overlap or lidar point density is insufficient at the canopy level.

### **Forest Mensuration from UAS-borne Point Clouds**

In contrast to the results of the quantitative checkpoints study and the qualitative CHM comparisons in Chapter 3, the application study highlighted a weakness of oblique imagery over the forested scene, demonstrating the superiority of the nadir CHM for individual tree detection

(ITD). Inspection of the CHMs and dense matching point clouds for both the nadir and oblique data show what is possibly two types of errors happening simultaneously which serves to benefit the nadir CHM as it undergoes local maxima detection (the ITD method used in Chapter 3).

1. The nadir data shows a tendency to underestimate canopy presence at a greater radius from the local maximum (i.e. treetop), as shown in Figures 3-9 and 3-11.
2. The oblique data demonstrates what is likely the false positive effect shown in Figure 3-1 in the direction of the optical axes of the cameras, which is no longer vertical, but rather inclined in the direction of flight (Figure 3-12, Tables 3-8 and 3-9).

Together, these effects lead to more distinct individual crowns in the nadir CHM. This is useful for ITD but may lead to underestimation of other metrics such as crown area or aboveground biomass. This false positive effect in the oblique data may be mitigated by either decreasing the angle of tilt or adding additional flight lines perpendicular to each other, though the latter option increases both flight and processing time. The intended use of the data should be considered before choosing to tilt the camera, if at all, and whether additional flight lines would provide a net benefit.

The stem detection results suggest that the augmented Maas *et al.* stem detection algorithm could be suitable for areas similar to those plots under study, i.e. homogeneous scenes where the only objects in the search area are tree stems, with limited understory growth. (Concurrent work has shown that this stem detection method is not as effective in a heterogeneous, urban scene.) It cannot be readily determined if the overestimation bias shown by the height estimation algorithm is due to inaccurate point cloud data or underestimation during the field cruise—a conceivable error when using a hypsometer to measure tree heights.

The algorithm for finding the DBH of individual stems shows promise of producing accurate results under certain conditions and with anticipated future improvements in sensor technology. The shape of the reconstructed tree stems in the dense matching point clouds is

highly irregular due to the geometry of the photos and the occlusions of the canopy; even using RANSAC, a model-fitting method suited for noisy data, the results so far have been highly imprecise and inaccurate. These irregularities are typically due to canopy occlusion, as the results from the baseline terrestrial study produced exceptional results. In the lidar point clouds, the lidar sensor in the airborne did not provide enough stem information for reliable cylinder fitting. The information available may also be less accurate due to the range-dependent error and large beam divergence of the VLP-16 scanner. With improvements in miniature GNSS/INS technology, this range-dependent error will decrease, leading to more relatively accurate features in the lidar point cloud.

### **Current State of the Art**

A stem-based approach to either ITD or stand-level sampling is currently not advantageous when compared to canopy-based estimation techniques under the conditions present in this study. As the accuracy of UAS-borne lidar sensors increases, sufficient, accurate information can be gathered about the individual stems of trees in a forested scene. Bear in mind that canopy-based methods of crown delineation are still prone to certain pitfalls, namely:

1. failure to detect trees below canopy,
2. failure to delineate when presented with overlapping crowns, and
3. sensitivity of tree count estimation with respect to the method of smoothing the CHM and distinctness of the tree crowns.

One of the greatest strengths of the UAS platform in remote sensing is how often it can be deployed. The high temporal resolution of UAS remote sensing data offers new insight into dynamic processes. Stem-based ITD is more effective as a forest matures and its trees become larger and more widely spaced. This is not necessarily a dynamic time in the life cycle of a tree, excepting external processes such as fire, disease, or logging. The most value in monitoring the

life cycle of a stand comes when the trees are younger, with smaller stems and typically closer spacing. Stem-based ITD is ineffective at this stage, and canopy-based ITD can only provide so much information about detected trees.

I am led to conclude that structural remote sensing from the UAS platform currently best serves forestry not as a potential tool for individual tree inventory, but as a “macrosampling” tool, capable of quickly and repeatedly sampling plots and delivering useful information at the plot level. Individual tree inventory from UAS remote sensing accurate to some arbitrarily acceptable value (say 5% or 10%) is not yet attainable for the typical Florida pine stand using stem-based methods. Canopy-based methods may provide this level of accuracy, but cannot offer individual stem information. As sensor technology improves, the goal of ITD with stem information may soon be attainable.

## APPENDIX A

### VLP-16 CHARACTERIZATION EQUATIONS

Note: The following equations and derivations include the work of Travis Whitley.

#### **Point Density Probability Density Function**

The first assumption that is presented is the angle of the laser firing from the lidar unit can be modeled as a uniform random variable  $\theta$ , i.e.  $\theta \sim U(-\pi, \pi)$ . This is valid since the lasers are being fired at a constant rate from the scanner head which is rotating at a constant angular velocity. The angle  $\theta$  is defined as rotation about the scanner's z-axis,  $Z_N$ , as depicted in Figure B-1. It is also assumed that the sensor is travelling parallel to the ground along its axis of rotation  $Z_N$ —i.e., at a constant height above ground—at a constant velocity  $v_Z$ . The frequency of the laser firings is designated as the variable  $l_f$ .

The cumulative distribution function (CDF) of  $\theta$  is:

$$F_\theta(\theta) = \begin{cases} 0 & \theta < -\pi/2 \\ \frac{\theta}{\pi} + \frac{1}{2} & \theta \in [-\pi/2, \pi/2] \\ 1 & \theta \geq \pi/2 \end{cases}$$

Note that this function limits  $\theta$  to the range  $[-\pi/2, \pi/2]$  because no pulses emitted outside of this range will intersect with the ground below (see Figure A-1).

The next step is to find the probability density function (PDF) of another random variable  $X$ , defined as the lateral distance of each laser return (in the  $X_N$  direction) from the centerline of the laser return pattern (i.e., the line directly nadir to the flight line of the scanner).  $X$  is a function of  $\theta$ , and can be defined as

$$X = h \tan \theta$$

where  $h$  is the height of the sensor above the ground. The CDF of  $X$  can then be defined as

$$F_X(x) = P(X \leq x)$$

Substituting yields

$$F_X(x) = P(h \tan \theta \leq x)$$

Because  $h$  is always positive:

$$F_X(x) = P\left(\tan \theta \leq \frac{x}{h}\right)$$

Inverse tangent is an increasing one-to-one function, which allows for further rearranging:

$$F_X(x) = P\left(\theta \leq \tan^{-1}\left(\frac{x}{h}\right)\right)$$

Substituting further yields

$$F_X(x) = P\left(\theta \leq \tan^{-1}\left(\frac{x}{h}\right)\right) = F_\theta\left(\tan^{-1}\left(\frac{x}{h}\right)\right)$$

$$F_X(x) = \frac{\tan^{-1}(x/h)}{\pi} + \frac{1}{2}$$

The derivative yields the function

$$f_X(x) = \frac{d}{dx} F_X(x) \Rightarrow$$

$$f_X(x) = \frac{h}{\pi(h^2 + x^2)}$$

To convert this PDF to a point density function as a function of  $x$ , or lateral distance from the centerline of the laser return pattern, the PDF is multiplied by the laser pulse frequency  $l_f$  and divided by the forward velocity  $v_z$ . (Note that  $l_f$  is divided by 2 because half of the laser firings will not intersect the ground.)

$$p(x) = \frac{l_f f_X(x)}{2v_z} \Rightarrow$$

$$p(x) = \frac{l_f h}{2\pi v_z (h^2 + x^2)}$$

## Optimal Separation of Parallel Flight Lines

With the equation for point density as a function of distance normal to the direction of flight, an equation can be derived that dictates the optimal separation distance between two consecutive, parallel flight lines (defined as  $w$ ) in order to achieve a desired minimum point density  $p_d$  in the overlapping laser return pattern. This minimum point density will occur exactly halfway between the two centerlines of the overlapping laser return patterns. (This can be proven, as  $p(x)$  is an even function, but this proof will not be presented here.) To restate the equation for  $p(x)$ :

$$p(x) = \frac{l_f h}{2\pi v_z (h^2 + x^2)}$$

Substituting the halfway point  $w/2$  for  $x$  and setting it equal to  $p_d/2$  yields

$$\frac{p_d}{2} = \frac{l_f h}{2\pi v_z (h^2 + (w/2)^2)}$$

Solving for  $w$  yields

$$w = 2 \sqrt{\frac{l_f h}{\pi p_d v_z} - h^2}$$

This equation can be used in conjunction with the percent overlap of the field of view of the sensor. To find the overlap, first the furthest lateral laser return is found using simple trigonometry:

$$x_{max} = \sqrt{l_m^2 - h^2}$$

where  $l_m$  is the maximum laser return range of the sensor. Once this is obtained, the percent overlap is defined as

$$\%_{overlap} = \frac{x_{max} - w}{x_{max}}$$

$$\%_{overlap} = 1 - \frac{w}{\sqrt{l_m^2 - h^2}}$$

Substituting for  $w$  yields

$$w(\%_{overlap}, l_m, h) = (1 - \%_{overlap})\sqrt{l_m^2 - h^2}$$

### Possible Positions of Coverage Gaps

As discussed in Chapter 2, gaps in coverage of laser returns along the ground can occur in strips parallel to the direction of flight. These gaps may occur when consecutive hyperbolic paths of points (from consecutive passes of the scanner head through the range of downward-facing azimuths  $[-\pi/2, \pi/2]$ ) intersect with each other, creating clusters of overlapping laser returns. These intersections will only occur where the separation between consecutive channels  $x$  equals some integer multiple  $i$  of the distance traveled by the scanner during one rotation of the scanner head, or

$$y = \frac{iv_z}{r}$$

where  $v_z$  is the forward velocity of the scanner and  $r$  is the rotation rate of the scanner head. The separation  $x$  will occur at some ranges from the scanner  $\ell_i$ , which can be expressed as

$$\ell_i = \frac{y}{\tan \Delta\omega} = \frac{iv_z}{r \tan \Delta\omega}$$

where  $\Delta\omega$  is the angular separation between channels (which, for the VLP-16, is  $2^\circ$ ). These ranges  $\ell_i$  occur at some angles from nadir to the scanner  $\alpha_i$ . With the scanner at flying height  $h$ , the lateral distance from the flight line  $y_i$  of the strips of coverage gaps can be expressed as

$$x_i = h \tan \alpha_i$$

Because of the relationship between  $h$  and  $\ell_i$  the equation above can be rewritten as

$$x_i = h \tan \left( \cos^{-1} \left( \frac{h}{\ell_i} \right) \right)$$

Substituting for  $\ell_i$  yields the gap equation

$$x_i = h \tan \left( \cos^{-1} \left( \frac{hr \tan \Delta\omega}{iv_z} \right) \right)$$

### Generalizing the Equations for Yaw

The preceding derivations assume that the scanner is oriented such that the z-axis of the scanner is parallel to the direction of travel to the scanner. In the case of yaw, the z-axis of the scanner is rotated at some angle  $\alpha$  with respect to the direction of travel. The following equations can be further generalized to account for the yaw of scanner with respect to the direction of travel.

### Point density probability function

The PDF of random variable  $X$  is defined as the lateral distance of each laser return (in the  $X_N$  direction) from the centerline of the laser return pattern.  $X$  is a function of  $\theta$  (Figure A-1) and the yaw angle  $\alpha$ , and can be defined as

$$X = h \tan \theta \cos \alpha$$

Thus the CDF of  $X$  can then be defined as

$$F_X(x) = P(h \tan \theta \cos \alpha \leq x)$$

Substituting and rearranging further yields

$$\begin{aligned} F_X(x) &= P \left( \theta \leq \tan^{-1} \left( \frac{x}{h \cos \alpha} \right) \right) = F_\theta \left( \tan^{-1} \left( \frac{x}{h \cos \alpha} \right) \right) \\ F_X(x) &= \frac{\tan^{-1}(x/(h \cos \alpha))}{\pi} + \frac{1}{2} \end{aligned}$$

The derivative yields the function

$$f_X(x) = \frac{h \cos \alpha}{\pi(h^2 \cos^2 \alpha + x^2)}$$

This is then converted to a PDF as a function of  $x$ , or lateral distance from the centerline of the laser return pattern:

$$p(x) = \frac{l_f f_X(x)}{2v_z} \Rightarrow$$

$$p(x) = \frac{l_f h \cos \alpha}{2\pi v_z (h^2 \cos^2 \alpha + x^2)}$$

### Optimal separation of flight lines

Substituting the halfway point  $w/2$  for  $x$  and setting it equal to  $p_d/2$ , where  $p_d$  is the desired minimum point density in the overlapping laser return pattern, yields

$$\frac{p_d}{2} = \frac{l_f h \cos \alpha}{2\pi v_z (h^2 \cos^2 \alpha + (w/2)^2)}$$

Solving for  $w$  yields

$$w = 2 \sqrt{\frac{l_f h \cos \alpha}{\pi p_d v_z} - h^2 \cos^2 \alpha}$$

### Gap equation

Because the solution of the gap position equation is stated in terms of the variable  $x$ , the equation can be generalized to account for yaw simply by multiplying the results by the cosine of the yaw angle:

$$x_i = x_i \cos \alpha \Rightarrow$$

$$x_i = h \tan \left( \cos^{-1} \left( \frac{hr \tan \Delta\omega}{iv_z} \right) \right) \cos \alpha$$

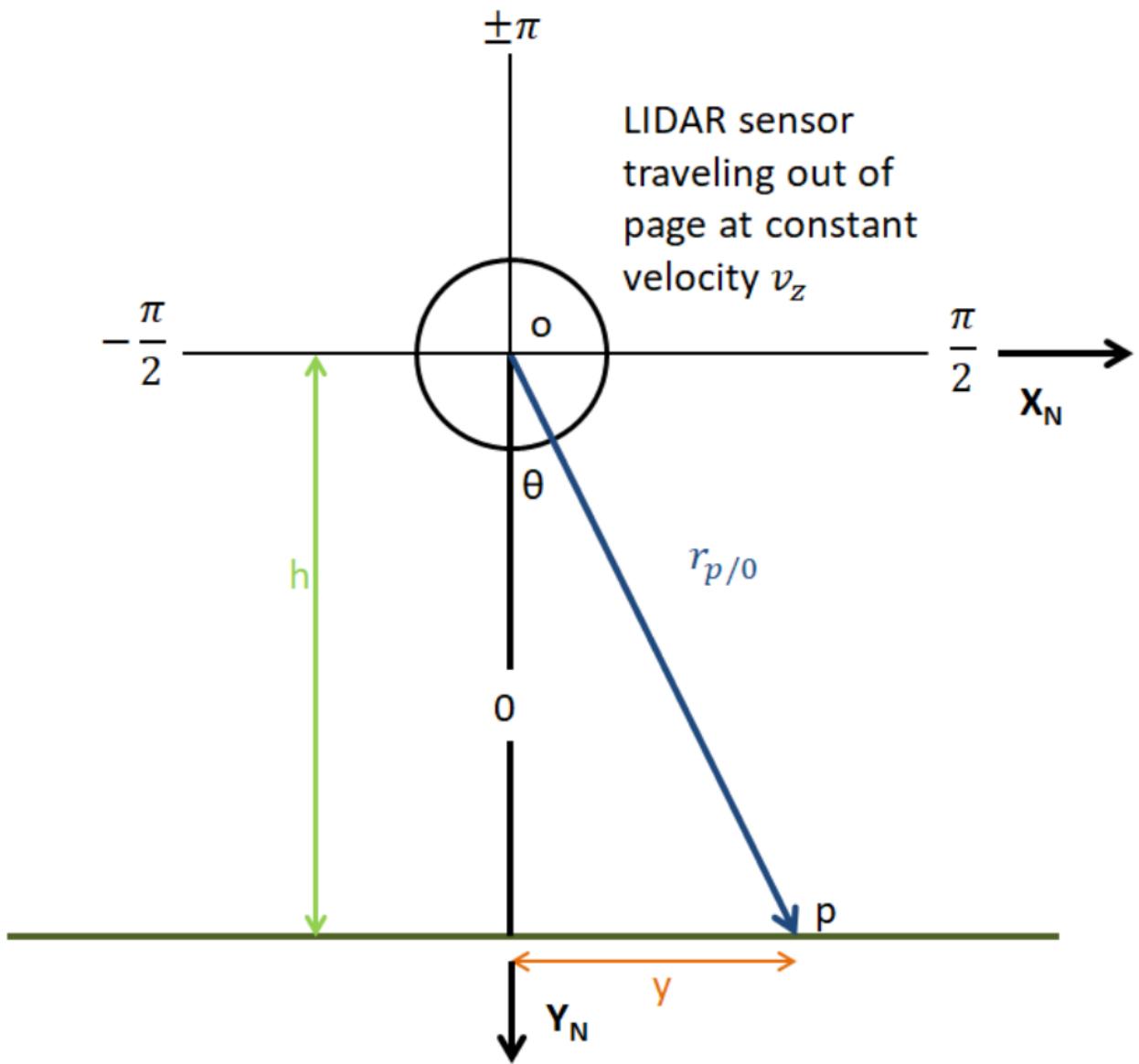


Figure A-1. Nominal configuration of VLP-16 in relation to ground.

## APPENDIX B

### VLP-16 SIMULATION SOFTWARE

#### Line-Plane Intersection

The VLP-16 simulation software models the laser return as an intersection of a line (laser pulse) and a plane (flat ground). The target plane is defined by a point on the plane  $\mathbf{p}_0 = (0,0,0)$  and a vector normal to the plane  $\mathbf{n} = \langle 0,0,1 \rangle$ . The software defines each laser pulse as a line, represented by a point—the location of the scanner at the time of firing,  $\mathbf{r}_s^m(t)$ —and a direction,  $\mathbf{v}_s(t)$ . (The term  $t$  is omitted below for simplicity.) This direction is initially represented by two angles, the azimuth  $\alpha$  and vertical angle  $\omega$ , both of which are defined in the scanner's local reference frame. The software first converts this direction to a vector in the scanner frame:

$$\mathbf{v}_s = \langle \cos \omega \sin \alpha, \cos \omega \cos \alpha, \sin \omega \rangle.$$

This vector is then rotated into the mapping frame. The orientation of the scanner is known in terms of tilt  $\tau$  and yaw  $\kappa$ ; these values are formed into rotation matrix  $\mathbf{R}_s^m$ , which is the product of sequential rotation about the scanner's x-axis and z-axis:

$$\begin{aligned}\mathbf{R}_s^m &= \mathbf{R}_\kappa \mathbf{R}_\tau = \begin{bmatrix} \cos \kappa & -\sin \kappa & 0 \\ \sin \kappa & \cos \kappa & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \tau & -\sin \tau \\ 0 & \sin \tau & \cos \tau \end{bmatrix} \Rightarrow \\ \mathbf{R}_s^m &= \begin{bmatrix} \cos \kappa & -\sin \kappa \cos \tau & \sin \kappa \cos \tau \\ \sin \kappa & \cos \kappa \cos \tau & -\cos \kappa \sin \tau \\ 0 & \sin \tau & \cos \tau \end{bmatrix}.\end{aligned}$$

The scanner space vector is then rotated into object space, and is held as the direction of the line:

$$\mathbf{v}_m = \mathbf{R}_s^m \mathbf{v}_s.$$

(Note that the subscript “s” denotes that this is the position vector of the scanner, and the superscript “m” denotes that this vector is defined in the mapping frame.) The intersection of the line and plane is found by the common algebraic method, which is not presented here.

## **Assumptions**

By default, the software models the VLP-16 scanner as a point moving parallel to “flat ground”—a target plane at a height of zero, with a normal parallel to the mapping frame +Z-axis. The laser returns are modeled by lines with a known direction passing through the scanner point and intersecting the target plane. The scanner is assumed to be oriented such that its local reference frame’s +z-axis is coincident with the direction of travel, which has been arbitrarily chosen to be the mapping frame +Y-axis. The scanner’s +y-axis is coincident with the -Z-axis in the mapping frame. Adjustments to the angular orientation of the scanner—namely, tilt and yaw—can be made as optional input arguments to the software. Because of the scanner’s 360° field of view, adjusting the roll would have practically no effect on the laser return pattern.

## **Optimization**

The VLP-16 emits approximately 300,000 pulses per second; from the aerial pose, however, less than half of those pulses will reach the ground below, and even fewer will be within the maximum range at which the scanner can reliably mark a pulse as a return. To speed up the simulation of the laser return pattern, more than half of the pulses that would be emitted by the VLP-16 need not be modeled.

The software can also produce a representative profile of the laser return pattern, in which the entire area of the profile has been “fully illuminated,” or has been struck by lasers from all sixteen channels. This representative profile is a repeating pattern; except for the beginning and end of a flight line, an entire single strip of laser returns exhibits this pattern. The scanner need only travel a certain distance forward before such a profile exists in the laser return pattern. Thus, the software only needs to simulate the laser return pattern for a short duration of time (on the order of seconds) before terminating.

## Azimuth bounds

The range of azimuths at which laser pulses will produce a return can be found via simple trigonometry:

$$\alpha_{max} = \cos^{-1}\left(\frac{h}{r_{max}}\right).$$

Under the default configuration, the scanner's +y-axis ( $\alpha = 0^\circ$ ) points directly nadir; thus, the range of azimuths for which line-plane intersections should be calculated is  $[-\alpha_{max}, \alpha_{max}]$ . Note that the maximum range of the VLP-16 has been found empirically to be 120 m, although the software does allow for manual adjustment to this value for the purposes of planning missions in which returns with ranges greater than a specified distance are eliminated (see Chapter 2).

Even if the scanner is oriented with nonzero tilt and/or yaw, this range of azimuths is valid. Assuming the VLP-16 had a spherical,  $360^\circ$  field of view in all planes, the radius of this sphere would be  $r_{max}$ . The intersection of this sphere with the target plane is a circle, the radius of which is  $\sqrt{r_{max}^2 - h^2}$ . The angle between the point on the target plane directly below the scanner, the scanner itself, and any point along the edge of this circle is equal to  $\alpha_{max}$ .

## Time Bounds

As mentioned earlier, the software can produce a representative profile of the laser return pattern. The scanner model only needs to travel a certain distance to create this profile. Because each second the modeled scanner spends “in flight” requires hundreds of thousands of line-plane intersections to be calculated, it is crucial to strictly limit the flight time in the simulation. To limit the time that elapses in the simulation, the exact distance the scanner must travel to create this representative profile must be found.

The representative profile's repeating pattern only occurs along those areas of the ground that receive pulses from all sixteen channels. The beginning of this repeating pattern is the

farthest point at which the most forward-facing channel ( $\omega = 15^\circ$ ) strikes the ground from the scanner's starting point, which will be called  $\mathbf{p}_{min}$ . The point  $\mathbf{p}_{min}$  is a function of the flying height  $h$ , tilt  $\tau$ , and yaw  $\kappa$ . The scanner, therefore, must travel forward until its most rear-facing channel ( $\omega = -15^\circ$ ) reaches  $\mathbf{p}_{min}$ .

The point  $\mathbf{p}_{min}$  can be found via line-plane intersection. For each line-plane intersection calculated, the software requires as input, among other variables, the azimuth of the laser firing. Thus, the maximum azimuth of the  $\omega = 15^\circ$  channel. This value is not calculated along a line perpendicular to the direction of travel and passing through a point nadir to the scanner, which were the two assumptions used to find  $\alpha_{max}$  previously; a more general formula must be derived.

The first assumption is to assume the location of the scanner is directly above the origin of the mapping frame at flying height:  $\mathbf{r}_s^m = \langle 0, 0, h \rangle$ . The scanner may be oriented with nonzero tilt and yaw, which can be represented by  $\mathbf{R}_s^m$ . It is assumed that some laser pulse  $\mathbf{r}_{\mathbf{p}_{min}}^s$  from the  $\omega = 15^\circ$  channel will reach the point  $\mathbf{p}_{min}$ . Thus  $\mathbf{p}_{min}$  can be expressed as

$$\mathbf{p}_{min} = (\mathbf{R}_s^m)^{-1} \mathbf{r}_{\mathbf{p}_{min}}^s + \mathbf{r}_s^m.$$

This problem is visualized in Figure A-1. (By the convention used in this paper, rotation matrices are used to rotate reference frames, or coordinate systems. For the problem of solving for  $\mathbf{p}_{min}$ , the goal is to solve for a value in the that is defined in the scanner frame,  $\alpha$ ; thus, the inverse of the rotation matrix must be used to rotate the mapping frame into the scanner frame. The inverse of the rotation matrix could alternately defined as  $\mathbf{R}_m^s = (\mathbf{R}_s^m)^{-1}$ .)

The equation above can be expanded as:

$$\mathbf{p}_{min} = \begin{bmatrix} \cos \kappa & -\sin \kappa \cos \tau & \sin \kappa \cos \tau \\ \sin \kappa & \cos \kappa \cos \tau & -\cos \kappa \sin \tau \\ 0 & \sin \tau & \cos \tau \end{bmatrix} \begin{bmatrix} r_{max} \cos \omega \sin \alpha \\ r_{max} \cos \omega \cos \alpha \\ r_{max} \sin \omega \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ h \end{bmatrix}.$$

Solving for  $\alpha$  can be simplified by limiting the solution to only the z-value of  $\mathbf{p}_{min}$ , which, because it is on the target plane, is equal to zero:

$$z_{\mathbf{p}_{min}} = 0 = [0 \quad \sin \tau \quad \cos \tau] \begin{bmatrix} r_{max} \cos \omega \sin \alpha \\ r_{max} \cos \omega \cos \alpha \\ r_{max} \sin \omega \end{bmatrix} + h$$

$$r_{max}(\sin \tau \cos \omega \sin \alpha + \cos \tau \sin \omega) = -h$$

$$\alpha = -\frac{h/r_{max} - \cos \tau \sin \omega}{\sin \tau \cos \omega}.$$

The simplification employed here results in an equation that is not a function of yaw. This is acceptable, because yaw has no effect on the orientation of a vector in the scanner frame. By virtue of this simplification, the scope of this equation is limited to solving for the azimuth  $\mathbf{r}_{\mathbf{p}_{min}}^s$ , which is the laser pulse that reaches  $\mathbf{p}_{min}$ .

As stated previously, to create a representative profile, the scanner must travel some distance  $d$  until its most rear-facing channel ( $\omega = -15^\circ$ ) reaches  $\mathbf{p}_{min}$ . The scanner must then travel from that point for the time it takes for the scanner head to complete one full rotation, in order to assure that the laser return pattern is repeated at least once. The symmetry of the scan pattern can be exploited in order to find  $d$ . If the most forward-reaching laser pulse reaches  $\mathbf{p}_{min}$ , then it follows that, if the scanner is directly above the origin as previously assumed, the most rear-facing laser pulse reaches  $-\mathbf{p}_{min}$ . The distance  $d$  is the difference between the y-components of  $\mathbf{p}_{min}$  and  $-\mathbf{p}_{min}$ :

$$d = 2y_{\mathbf{p}_{min}}.$$

The length of the repeating pattern is a function of forward speed; but at most sensible speeds, this length is quite short, on the order of decimeters. For the sake of making the pattern more visible to a human user, the software allows for the representative profile to be of a predefined length  $l$ , usually on the order of 3-5 m. Thus, the software simulates the scanner traveling a

distance of  $d + l$ , and only returns those points within the bounds of the representative profile.

Because the simulation begins with the scanner above the origin and traveling in the +Y-

direction in the mapping frame, the points inside the profile are those with Y-coordinates

$$d < y_{points} \leq d + l.$$

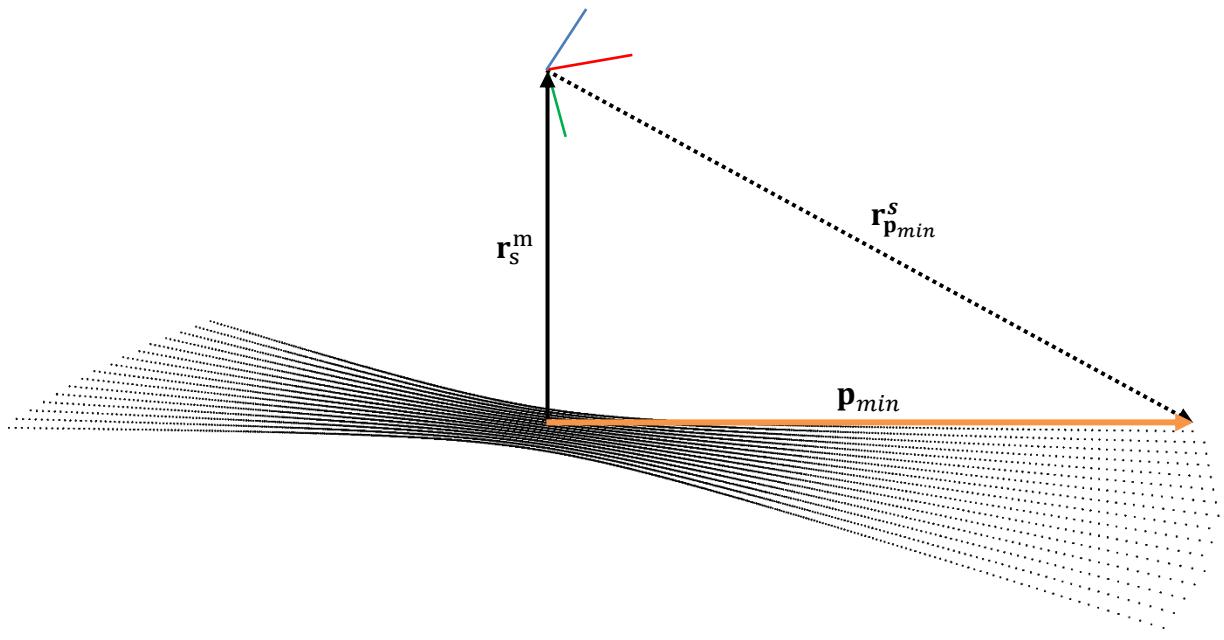


Figure B-1. Solving for the representative profile. The scanner frame's x-, y-, and z-axes are shown in red, green, and blue, respectively.

APPENDIX C  
DERIVATION OF OBLIQUE UAS PHOTOGRAMMETRY EQUATIONS

**Air Base and Air Width**

The length of the extent of coverage of an image, or footprint  $B_f$ , in the direction of the tilt of the camera can be found as follows:

$$\begin{aligned}x' &= H' \tan\left(t + \frac{\phi}{2}\right) \\x'' &= -H' \tan\left(t - \frac{\phi}{2}\right) \\B_f &= x' + x'' \quad \Rightarrow \\B_f &= H' \left[ \tan\left(t + \frac{\phi}{2}\right) - \tan\left(t - \frac{\phi}{2}\right) \right]\end{aligned}$$

In the above equations, flying height is denoted as  $H'$ , field of view in the direction of flight at  $\phi$ , tilt of the camera from nadir as  $t$ , and the values  $x'$  and  $x''$  are intermediate values (Figure C-1). The air base for the exposure stations is expressed as a percentage of the length of the footprint in the direction of flight, which is reflected in the equation as presented in Chapter 3.

The width of the footprint  $W_f$  is variable in the direction of travel of the camera. The footprint of a rectangular format camera, when tilted with respect to the target plane, will be in the shape of a trapezoid. For mission planning purposes, the width of the footprint at a line intersecting and normal to the optical axis is used:

$$\begin{aligned}\frac{w}{f} &= \frac{W_f}{H'/\cos t} \quad \Rightarrow \\W_f &= H' \frac{w}{f \cos t}\end{aligned}$$

where  $f$  is the focal length. Similarly, the air width for the exposure stations is expressed as a percentage of the length of the footprint normal to the direction of flight, as expressed in the equation as presented in Chapter 3.

### Ground Sample Distance

The ground sample distance (GSD) of a pixel in column  $p$  is variable along the direction of tilt. For  $m$  columns of pixels, the GSD of a pixel  $p$  in the direction of tilt  $x$ , or  $GSD_{x,p}$  is expressed by:

$$GSD_{x,p} = \gamma' - \gamma''$$

$$\gamma' = H' \tan(t + \mu_{p+1})$$

$$\gamma'' = H' \tan(t + \mu_p)$$

$$GSD_{x,p} = H' [\tan(t + \mu_{p+1}) - \tan(t + \mu_p)]$$

$$\mu_p = \text{atan}\left(\frac{p - m/2}{f}\right).$$

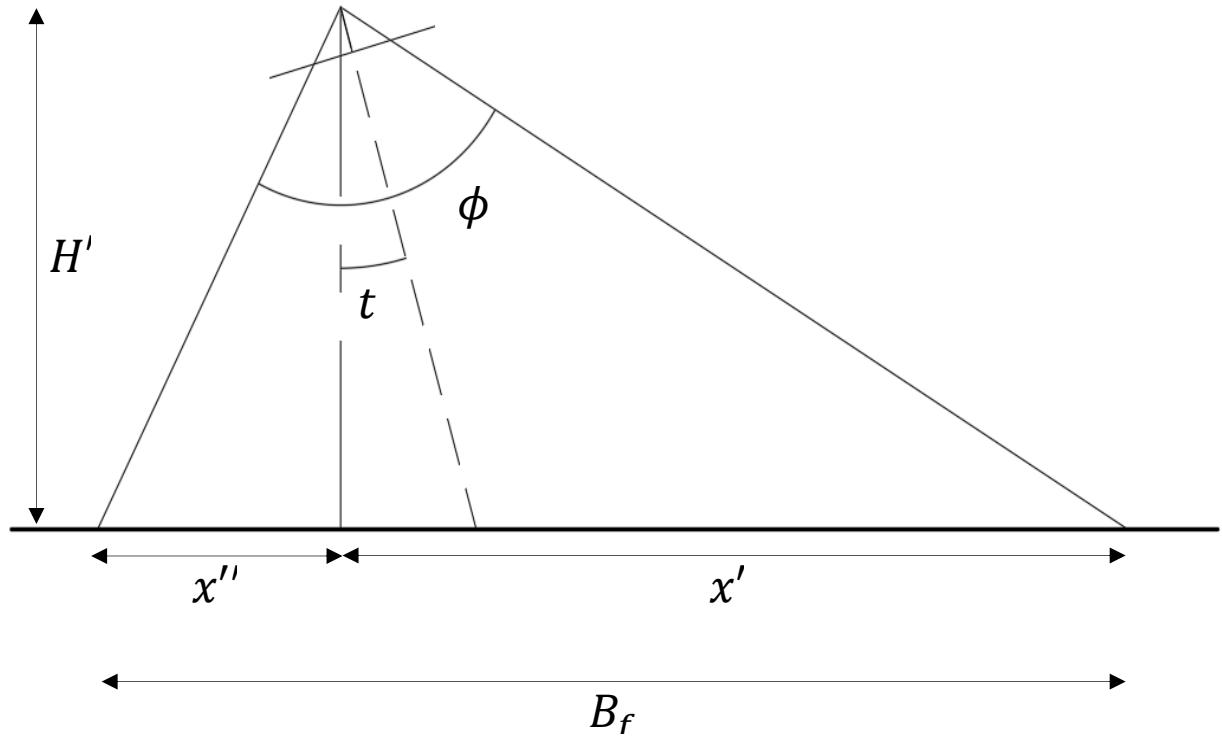


Figure C-1. Air base.

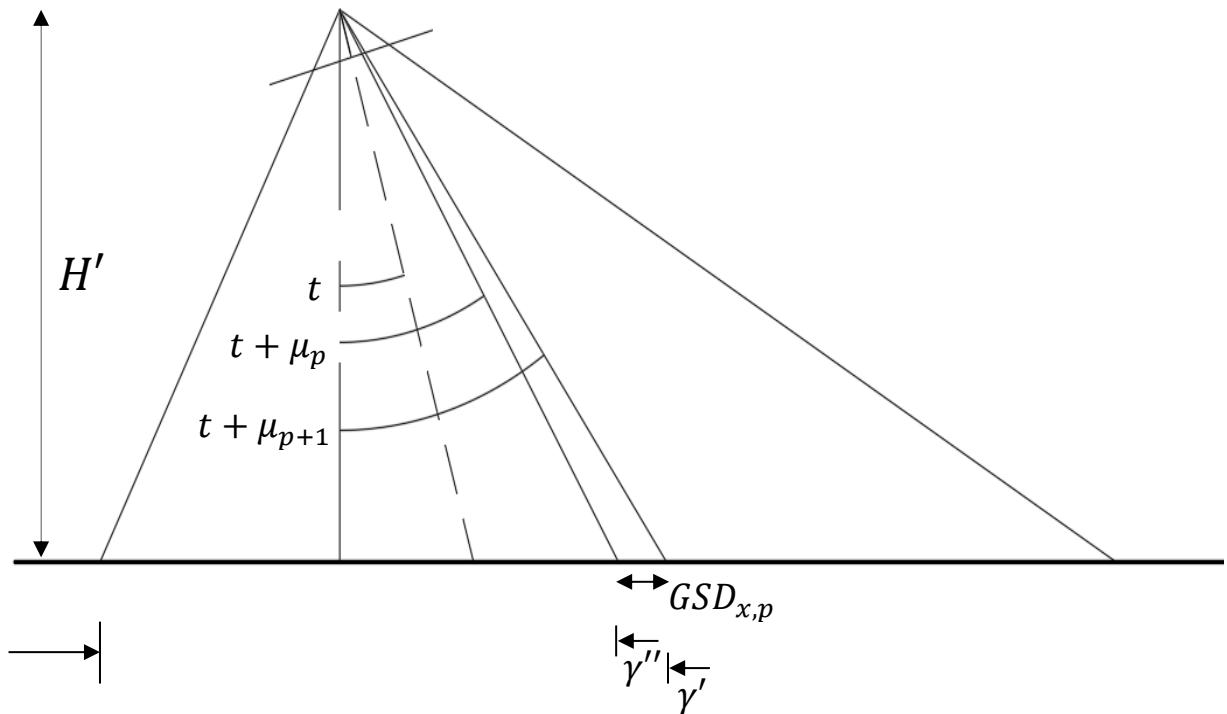


Figure C-2. Ground sample distance in the direction of tilt.

## APPENDIX D SIMULATING IMAGES IN MATLAB

One facet of the UAS photogrammetric mission planning software described in Chapter 4 involves creating simulated images of a simple test scene in a simulated object space. Each image is the projection of this test scene—an array of 3D points—onto the focal plane of a virtual camera. The position and angular attitude—i.e., the external orientation parameters, or EOPs—of each camera is defined by the mission parameters of flying height, tilt of camera from nadir, overlap, and side lap. (Overlap and side lap are a function of the camera’s field of view, which itself is a function of focal length and size of format.) In other words, the position and orientation of each virtual focal plane is known with respect to the positions of the 3D array in the test scene. The position of a point in the test scene on the virtual image is where the line between the test scene point and the camera’s focal point intersects the focal plane. This alignment satisfies the collinearity condition; solving for these positions, therefore, can be achieved using the collinearity equations. Furthermore, the collinearity equations will be used to transform coordinates from object space to image space, so using a homogeneous representation of the collinearity equations will simplify the problem (Wolf et al. 2014).

The collinearity equations as stated by Wolf et al. (2014) are

$$x_a = -f \left[ \frac{r_{11}(X_A - X_L) + r_{12}(Y_A - Y_L) + r_{13}(Z_A - Z_L)}{r_{31}(X_A - X_L) + r_{32}(Y_A - Y_L) + r_{33}(Z_A - Z_L)} \right]$$

$$y_a = -f \left[ \frac{r_{21}(X_A - X_L) + r_{22}(Y_A - Y_L) + r_{23}(Z_A - Z_L)}{r_{31}(X_A - X_L) + r_{32}(Y_A - Y_L) + r_{33}(Z_A - Z_L)} \right]$$

These equations can be represented homogeneously as

$$\tilde{\mathbf{x}} = \mathbf{PRTX} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1/f & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -X_L \\ 0 & 1 & 0 & -Y_L \\ 0 & 0 & 1 & -Z_L \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_A \\ Y_A \\ Z_A \\ 1 \end{bmatrix}$$

where  $\tilde{\mathbf{x}}$  are the homogeneous  $(wx, wy, w)$  for point  $(x, y)$  in image space;  $\mathbf{P}$  is the perspective projection transformation matrix, which projects the homogeneous point onto the  $-f$  focal plane in image space;  $\mathbf{R}$  is the rotation matrix, which represents the rotation of object space into image space; the translation matrix  $\mathbf{T}$ , which represents the position of the focal point  $L$  in object space; and the vector  $\mathbf{X}$ , which contains the coordinates for the object space point being projected.

Although  $\tilde{\mathbf{x}}$  and  $\mathbf{X}$  above are represented as column vectors, they can in fact be  $(3 \times n)$  and  $(4 \times n)$  matrices, respectively, consisting of  $n$  concatenated column vectors. Thus, multiple points in object space can be projected onto a single focal plane at once.

Simulating the images in MATLAB thus requires as input the 3D, object space coordinates of the points in the test scene, the EOPs of each camera, and the focal length  $f$  (in pixels) of the simulated camera. The projection is solved using the method described above, and the results in matrix  $\tilde{\mathbf{x}}$  represent the image space coordinates on the image place  $z = w$ . The final image space coordinates  $\mathbf{x}$  (i.e., the image space coordinates on plane  $-f$ ) are found by

$$\mathbf{x} = \frac{1}{w} \tilde{\mathbf{x}}$$

where the values of  $\mathbf{x}$  are in pixels.

The image simulation also requires the size of the camera format (in pixels), so that all points projected into a camera's image space that fall outside the bounds of the format size can be omitted from the image. Also, as stated in Chapter 3, random noise is added to the image projections. The noise values are pseudorandom numbers drawn from a standard Gaussian distribution, multiplied by  $\sigma = 0.5$ , to simulate noise of 0.5 pixel.

## APPENDIX E CONVERSION OF ROTATION DATA BETWEEN ENU AND NED FRAMES

As stated in the Conclusion, the rotation data reported by the Xsens Mt-G-710 is not directly useable as input in Agisoft PhotoScan. A conversion between the rotation angles reported by the Xsens in the Easting-Northing-Up (ENU) frame and the rotations expected by PhotoScan in the Northing-Easting-Down (NED) frame is presented. Note that all coordinate systems presented here are right-hand Cartesian systems. The convention used for camera orientation is such that the image  $+x$ -axis runs from left to right wing of the aircraft; the image  $+y$ -axis runs from tail to nose; and the  $-z$ -axis points downward perpendicular to the  $xy$ -plane.

PhotoScan accepts input for rotation angles in one of the following formats.

1. Omega-phi-kappa angles ( $\omega, \phi, \kappa$ ): sequential counterclockwise (CCW) rotations about the  $+x$ -,  $+y$ -, and  $+z$ -axes, respectively, of the image, where an image taken at  $(\omega, \phi, \kappa) = (0^\circ, 0^\circ, 0^\circ)$  results in a vertical (nadir) image oriented such that its horizontal axis is parallel with the easting axis of the local tangential plane (LTP) and its vertical axis is parallel with LTP northing. Put another way, the rotations  $\omega, \phi, \kappa$  are defined with respect to the ENU frame, which is parallel with the basis of the LTP.
2. Yaw-pitch-roll angles ( $y, p, r$ ): sequential, CCW rotations about the down, easting, and northing axes of the airframe, where an image taken at  $(y, p, r) = (0^\circ, 0^\circ, 0^\circ)$  indicates an image oriented as described above. Put differently, the rotations  $ypr$  are defined with respect to the Northing-Easting-Down (NED) frame.

(The following definitions are found in the PhotoScan 1.4 User Manual (2018). Due to the configuration of the Xsens used during image collection, the rotation angles provided for each image are called yaw, pitch, and roll, but are defined differently than above. The Xsens standard defines yaw-pitch-roll as sequential CCW rotations about the  $+z$ -,  $+y$ , and  $+x$ -axes; these rotations are referred to hereafter as  $(y', p', r')$ . Also by the Xsens standard, the  $+x$ -axis, or roll axis, of the unit should point from tail to nose of the aircraft. Thus, the orientation of an image parallel with the LTP is reported as  $(y', p', r') = (90^\circ, 0^\circ, 0^\circ)$  (Xsens, 2018).

Conversion from  $(y', p', r')$  to  $(\omega, \phi, \kappa)$  or  $(y, p, r)$  is possible if a rotation matrix is defined for each system. To facilitate the conversion, the rotation matrix for each system will be defined in the ENU frame.

### Rotation Matrix from Yaw-Pitch-Roll, Xsens Definition

The construction of a rotation matrix for  $(y', p', r')$  must account for the convention of the roll axis of the unit being parallel with the  $+y$ -axis of the image. This is achieved by constructing the rotation matrix using the reported values for yaw, pitch, and roll, and then rotating this rotation matrix by  $-90^\circ$  about the twice-rotated  $+z$ -axis,  $\mathbf{R}_{z'', -90^\circ}$ . The rotation matrix is a series of transpose Givens rotations. (Givens rotations as commonly defined rotate vectors in space; the transpose will effectively rotate the space itself, which is the convention in photogrammetry.) For brevity, cosine and sine of some angle  $\theta$  are denoted as  $c_\theta$  and  $s_\theta$ , respectively.

$$\begin{aligned}
 \mathbf{R}_{ENU}(y', p', r') &= \mathbf{R}_{z'', -90^\circ} \mathbf{R}_{r'} \mathbf{R}_{p'} \mathbf{R}_{y'} \\
 &= \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_{r'} & s_{r'} \\ 0 & -s_{r'} & c_{r'} \end{bmatrix} \begin{bmatrix} c_{p'} & 0 & -s_{p'} \\ 0 & 1 & 0 \\ s_{p'} & 0 & c_{p'} \end{bmatrix} \begin{bmatrix} c_{y'} & s_{y'} & 0 \\ -s_{y'} & c_{y'} & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_{r'} & s_{r'} \\ 0 & -s_{r'} & c_{r'} \end{bmatrix} \begin{bmatrix} c_{p'} c_{y'} & c_{p'} s_{y'} & -s_{p'} \\ -s_{y'} & c_{y'} & 0 \\ s_{p'} c_{y'} & s_{p'} s_{y'} & c_{p'} \end{bmatrix} \\
 &= \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_{p'} c_{y'} & c_{p'} s_{y'} & -s_{p'} \\ -c_{r'} s_{y'} + s_{r'} s_{p'} c_{y'} & c_{r'} c_{y'} + s_{r'} s_{p'} s_{y'} & s_{r'} c_{p'} \\ s_{r'} s_{y'} + c_{r'} s_{p'} c_{y'} & -s_{r'} c_{y'} + c_{r'} s_{p'} s_{y'} & c_{r'} c_{p'} \end{bmatrix} \\
 \mathbf{R}_{ENU}(y, p, r) &= \begin{bmatrix} c_{r'} s_{y'} - s_{r'} s_{p'} c_{y'} & c_{r'} c_{y'} - s_{r'} s_{p'} s_{y'} & -s_{r'} c_{p'} \\ c_{p'} c_{y'} & c_{p'} s_{y'} & -s_{p'} \\ s_{r'} s_{y'} + c_{r'} s_{p'} c_{y'} & -s_{r'} c_{y'} + c_{r'} s_{p'} s_{y'} & c_{r'} c_{p'} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}
 \end{aligned}$$

The values of the omega-phi-kappa ( $\omega, \phi, \kappa$ ) rotation angles can be found from this matrix using the definitions provided by Wolf et al. (2014):

$$\phi = \sin^{-1}(r_{31})$$

$$\omega = \tan^{-1}\left(\frac{-r_{32}}{r_{33}}\right)$$

$$\kappa = \tan^{-1}\left(\frac{-r_{21}}{r_{11}}\right)$$

For the special case of gimbal lock, in which  $|\phi| = 90^\circ$ , the twice-rotated  $x$ -axis becomes either parallel or antiparallel to the once-rotated  $z$ -axis, which would allow for ambiguous solutions. In the case of gimbal lock,  $\omega = 0^\circ$  is held, and

$$\kappa = \tan^{-1}\left(\frac{r_{12}}{r_{22}}\right)$$

### Rotation from Yaw-Pitch-Roll, Aviation Convention

Yaw-pitch-roll is more commonly defined in the NED frame, as described in definition #2 above. To construct a rotation matrix in the ENU frame from  $(y, p, r)$  angles, the orthogonal relationship of the NED and ENU frames can be exploited to form the alternate definition of  $(y, p, r)$  as sequential, CCW rotations about the  $-z$ -,  $+x$ -, and  $+y$ -axes of the image. By alternately defining a CCW rotation about the  $-z$ -axis as a clockwise (CW) rotation about the  $+z$ -axis, the rotation matrix can be constructed as follows:

$$\mathbf{R}_{ENU}(y, p, r) = \begin{bmatrix} c_r & 0 & -s_r \\ 0 & 1 & 0 \\ s_r & 0 & c_r \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_p & s_p \\ 0 & -s_p & c_p \end{bmatrix} \begin{bmatrix} c_y & -s_y & 0 \\ s_y & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} c_r & 0 & -s_r \\ 0 & 1 & 0 \\ s_r & 0 & c_r \end{bmatrix} \begin{bmatrix} c_y & -s_y & 0 \\ c_p s_y & c_p c_y & s_p \\ -s_p s_y & -s_p c_y & c_p \end{bmatrix}$$

$$\mathbf{R}_{ENU}(y, p, r) = \begin{bmatrix} c_r c_y + s_r s_p s_y & -c_r s_y + s_r s_p c_y & -s_r c_p \\ c_p s_y & c_p c_y & s_p \\ s_r c_y - c_r s_p s_y & -s_r s_y - c_r s_p c_y & c_r c_p \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

From  $\mathbf{R}_{ENU}(y, p, r)$  the original values for  $(y, p, r)$  can be found by

$$y = \tan^{-1} \left( \frac{r_{21}}{r_{22}} \right)$$

$$p = \sin^{-1}(r_{23})$$

$$r = \tan^{-1} \left( \frac{-r_{13}}{r_{33}} \right)$$

In the case of gimbal lock, where  $|p| = 90^\circ$ ,  $y = 0^\circ$  is held, and

$$r = \tan^{-1} \left( \frac{r_{31}}{r_{11}} \right)$$

The following is presented as a proof of a simpler relationship between the Xsens  $(y', p', r')$  and the conventional  $(y, p, r)$ . Through substitution it can be shown that

$$y = \tan^{-1} \left( \frac{r_{21}}{r_{22}} \right) = \tan^{-1} \left( \frac{c_{p'} c_{y'}}{c_{p'} s_{y'}} \right) = 90^\circ - y'$$

$$p = \sin^{-1}(r_{23}) = \sin^{-1}(-s_{p'}) = -p'$$

$$r = \tan^{-1} \left( \frac{-r_{13}}{r_{33}} \right) = \tan^{-1} \left( \frac{s_{r'} c_{p'}}{c_{r'} c_{p'}} \right) = r'$$

This simple algebraic conversion was used in Chapter 3 to convert the reported rotation angles of the Xsens Mt-G-710 from the Xsens convention to the PhotoScan convention.

## LIST OF REFERENCES

- Agisoft, 2018. Agisoft PhotoScan user manual, professional edition, version 1.4. Available: [http://www.agisoft.com/pdf/photoscan-pro\\_1\\_4\\_en.pdf](http://www.agisoft.com/pdf/photoscan-pro_1_4_en.pdf) (last date accessed: 02 July 2018).
- Avery, G., 1958. Helicopter stereo-photography of forest plots. *Photogrammetric Engineering*, Vol. 24, no. 4, pp. 617–624.
- Baltsavias, E.P., 1999. Airborne laser scanning: basic relations and formulas. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 54, pp. 199–214.
- Bohlin, J., J. Wallerman and J.E.S. Fransson, 2012. Forest variable estimation using photogrammetric matching of digital aerial images in combination with a high-resolution DEM. *Scandinavian Journal of Forest Research*, Vol. 27, pp. 692–699.
- Clark, P.J. and F.C. Evans, 1954. Distance to nearest neighbor as a measure of spatial relationships in populations. *Ecology*, Vol. 35, pp. 445–453.
- Dandois, J.P. and E.C. Ellis, 2013. High spatial resolution three-dimensional mapping of vegetation spectral dynamics using computer vision. *Remote Sensing of Environment*, Vol. 136, pp. 259–276.
- Dhondt, S., N. Wuyts and D. Inzé, 2013. Cell to whole-plant phenotyping: the best is yet to come. *Trends in Plant Science*, Vol. 18, pp. 428–439.
- El-Sheimy, N., 2008. Georeferencing component of LiDAR systems. In *Topographic Laser Ranging and Scanning*, CRC Press, Boca Raton, FL.
- Fischler, M.A. and R.C. Bolles, 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, Vol. 24, pp. 381–395.
- Fonstad, M.A., J.T. Dietrich, B.C. Courville, J.L. Jensen and P.E. Carbonneau, 2013. Topographic structure from motion: a new development in photogrammetric measurement. *Earth Surface Processes and Landforms*, Vol. 38, no. 4, pp. 421–430.
- Fritz, A., T. Kattenborn and B. Koch, 2013. UAV-based photogrammetric point clouds - tree stem mapping in open stands in comparison to terrestrial laser scanner point clouds. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 40, part 1/W2, pp. 141–146.
- Gagnon, P.A., J.P. Agnard and C. Nolette, 1993. Evaluation of a soft-copy photogrammetry system for tree-plot measurements. *Canadian Journal of Forest Research*, Vol. 23, no. 9, pp. 1781–1785.
- Glennie, C. and D.D. Lichti, 2010. Static calibration and analysis of the Velodyne HDL-64E S2 for high accuracy mobile scanning. *Remote Sensing*, Vol. 2, pp. 1610–1624.

Hartley, R. and A. Zimmerman, 2003. Multiple view geometry in computer vision. Cambridge University Press, New York.

Isenburg, M., 2012. LAStools—efficient tools for LiDAR processing. Available: <http://www.cs.unc.edu/~isenburg/lastools/> (last date accessed: 02 June 2018).

James, M.R. and S. Robson, 2014. Mitigating systematic error in topographic models derived from UAV and ground-based image networks. *Earth Surface Processes and Landforms*, Vol. 39, pp. 1413–1420.

Kankare, V., M. Holopainen, M. Vastaranta, E. Puttonen, X. Yu, J. Hyppä, M. Vaaja, H. Hyppä and P. Alho, 2013. Individual tree biomass estimation using terrestrial laser scanning. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 75, pp. 64–75.

Karpina, M., M. Jarząbek-Rychard, P. Tymków and A. Borkowski, 2016. UAV-based automatic tree growth measurement for biomass estimation. *Proceedings of the 23<sup>rd</sup> ISPRS Congress*, 12-19 July, Prague, Czech Republic, Vol. 41, part B8, pp. 685-688.

Khosravipour, A., A.K. Skidmore, M. Isenburg, T. Wang, and Y.A. Hussin, 2014. Generative pit-free canopy height models from airborne lidar. *Photogrammetric Engineering & Remote Sensing*, Vol. 80, no. 9, pp. 863-872.

Király, G. and G. Brolly, 2007. Tree height estimation methods for terrestrial laser scanning in a forest reserve. *International Archives of Photogrammetry and Remote Sensing*, Vol. 36, part 3/W52, pp. 211–215.

Korpela, I., B. Dahlin, H. Schäfer, E. Bruun, F. Haapaniemi, J. Honkasalo, S. Ilvesniemi, V. Kuuti, M. Linkosalmi, J. Mustonen, M. Salo, O. Suomi and H. Virtanen, 2007. Single-tree forest inventory using LiDAR and aerial images for 3D treetop positioning, species recognition, height and crown width estimation. *International Archives of Photogrammetry and Remote Sensing*, Vol. 36, part 3/W52, pp. 227-233.

Limentani, G.B., M.C. Ringo, F. Ye, M.L. Bergquist and E.O. McSorley, 2005. Beyond the t-test: statistical equivalence testing. *Analytical Chemistry*, Vol. 77, pp. 221A–226A.

Li, W., Q. Guo, M.K. Jakubowski, and M. Kelly, 2012. A new method for segmenting individual trees from the lidar point cloud. *Photogrammetric Engineering & Remote Sensing*, Vol. 78, pp. 75-84.

Lin, Y., J. Hyppä and A. Jaakkola, 2011. Mini-UAV-borne lidar for fine-scale mapping. *IEEE Geoscience and Remote Sensing Letters*, Vol. 8, pp. 426–430.

Lingua, A., D. Marenchino and F. Nex, 2009. Performance analysis of the SIFT operator for automatic feature extraction and matching in photogrammetric applications. *Sensors*, Vol. 9, pp. 3745–3766.

- Lisein, J., M. Pierrot-Deseilligny, S. Bonnet, and P. Lejeune, 2013. A photogrammetric workflow for the creation of a forest canopy height model from small unmanned aerial system imagery. *Forests*, Vol. 4, pp. 922–944.
- Lyons, E.H., 1966. Fixed air-base 70 mm photography, a new tool for forest sampling. *The Forestry Chronicle*, Vol. 42, pp. 420–431.
- Maas, H.G., A. Bienert, S. Scheller, and E. Keane, 2008. Automatic forest inventory parameter determination from terrestrial laser scanner data. *International Journal of Remote Sensing*, Vol. 29, pp. 1579–1593.
- Mohan, M., C.A. Silva, C. Klauberg, P. Jat, G. Catts, A. Cardill, A.T. Hudak and M. Dia, 2017. Individual tree detection from unmanned aerial vehicle (UAV) derived canopy height model in an open canopy mixed conifer forest. *Forests*, Vol. 8, no. 9, p. 340.
- Moskal, L.M. and G. Zheng, 2011. Retrieving forest inventory variables with terrestrial laser scanning (TLS) in urban heterogeneous forest. *Remote Sensing*, Vol. 4, pp. 1–20.
- Nilsson, M., 1996. Estimation of tree heights and stand volume using an airborne LiDAR system. *Remote Sensing of Environment*, Vol. 56, pp 1-7.
- Nex, F. and Remondino, 2014. UAV for 3D mapping applications: a review. *Applied Geomatics*, Vol. 6, pp. 1–15.
- O’Sullivan, D. and D.J. Unwin, 2010. Geographic Information Analysis. John Wiley & Sons, Hoboken, New Jersey.
- Pilarska, M., W. Ostrowski, K. Bakuła, K. Górska and Z. Kurczyński, 2016. The potential of light laser scanners developed for unmanned aerial vehicles—the review and accuracy. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 62, part 2/W2, pp. 87-95.
- Reece, K., 2014. Step-stare oblique aerial camera system. United States patent no. US8687062 B1.
- Reitberger, J., Cl. Schnörr, P. Krystek, and U. Stilla, 2009. 3D segmentation of single trees exploiting full waveform LIDAR data. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 64, pp. 561-574.
- Ring, J., 1963. The Laser in Astronomy. *New Scientist*, Vol. 18, no. 344, pp. 672-673.
- Schnabel, R., R. Wahl, and R. Klein, 2007. Efficient RANSAC for point-cloud shape detection. *Computer Graphics Forum*, Vol. 26, pp. 214–226.
- Schultz, S.L, F.D. Giuffrida and R.L. Gray, 2011. Mosaic oblique images and methods of making and using same. United States patent no. US7873238 B2.

- Schultz, S.L., F.D. Giuffrida, F.D., R.L. Gray and C. Mondello, 2010. Method and apparatus for capturing, geolocating and measuring oblique images. United States patent no. US7787659 B2.
- Silva, C.A., N.L Crookston, A.T. Hudak, and L.A. Vierling, 2017. rLiDAR: an R package for reading, processing and visualizing LiDAR (light detection and ranging) data. Available: <http://cran.rproject.org/web/packages/rLiDAR/index.html> (last accessed 05 July 2018).
- Spurr, S.H., 1948. *Aerial photographs in forestry*. The Ronald Press Company, New York.
- St-Onge, B.A. and N. Achaichia, 2001. Measuring forest canopy height using a combination of LiDAR and aerial photography data. *International Archives of Photogrammetry and Remote Sensing*, Vol. 34, part 3/W4, pp. 131–137.
- Starek, M. and J. Jung, 2015. The state of lidar for UAS applications. Lidar's next geospatial frontier. *2015 UAS Special of GIM International*, pp. 25–27.
- Suárez, J.C., C. Ontiveros, S. Smith and S. Snape, 2005. Use of airborne LiDAR and aerial photography in the estimation of individual tree heights in forestry. *Computers & Geosciences*, Vol. 31, pp. 253–262.
- Tang, L. and G. Shao, 2015. Drone remote sensing for forestry research and practices. *Journal of Forestry Research*, Vol. 26, pp. 791–797.
- USGS National UAS Project Office, 2017. Unmanned aircraft systems data post-processing: structure-from-motion photogrammetry. Oral presentation. *UAS Federal Users Workshop*, 28-30 March, Mountain View, CA.
- van Laar, A. and A. Akça, 2007. *Forest mensuration*. Springer, Dordrecht, Netherlands.
- Velodyne LiDAR., 2014. VLP-16 user's manual and programming Guide. Available: <http://velodynelidar.com/vlp-16.html> (last date accessed: 02 June 2018).
- Wang, L., P. Gong, and G.S. Biging, 2004. Individual tree-crown delineation and treetop detection in high-spatial-resolution aerial imagery. *Photogrammetric Engineering & Remote Sensing*, Vol. 70, no. 3, pp. 351-357.
- Watt, P.J. and D.N.M. Donoghue, 2005. Measuring forest structure with terrestrial laser scanning. *International Journal of Remote Sensing*, Vol. 26, pp. 1437–1446.
- Watts, A.C., V.G. Ambrosia, and E.A. Hinkley, 2012. Unmanned aircraft systems in remote sensing and scientific research: classification and considerations of use. *Remote Sensing*, Vol. 4, pp. 1671–1692.
- Westoby, M.J., J. Brasington, N.F. Glasser, M.J. Hambrey and J.M. Reynolds, 2012. “Structure-from-motion” photogrammetry: a low-cost, effective tool for geoscience applications. *Geomorphology*, Vol. 179, pp. 300–314.

Wolf, P.R., B.A. Dewitt and B.E. Wilkinson, 2014. *Elements of photogrammetry: with applications in GIS*, 4th ed. McGraw Hill, New York.

Xsens Technologies, 2018. MTi user manual. Available:  
[http://www.xsens.com/download/usermanual/MTi\\_usermanual.pdf](http://www.xsens.com/download/usermanual/MTi_usermanual.pdf) (last date accessed: 02 July 2018).

Yilmaz, V., C. Serifoglu and O. Gungor, 2016. Determining stand parameters from UAS-based point clouds. *Proceedings of the 23<sup>rd</sup> ISPRS Congress*, 12-19 July, Prague, Czech Republic, Vol. 41, part B8, pp. 413-416.

Zarco-Tejada, P.J., R. Diaz-Varela, V. Angileri and P. Loudjani, 2014. Tree height quantification using very high resolution imagery acquired from an unmanned aerial vehicle (UAV) and automatic 3D photo-reconstruction methods. *European Journal of Agronomy*, Vol. 55, pp. 89–99.

## BIOGRAPHICAL SKETCH

H. Andrew Lassiter received a Ph.D. in forest resources and conservation, with a specialty in geomatics, from the University of Florida in 2018. His research focuses on mission planning for UAS remote sensing, particularly the development of simulation software for both lidar and structure-from-motion (SfM) photogrammetric data collection, and quality assessment of data collected from different sensor configurations. He has also conducted research in automated and semi-automated estimation of key traits of forest stands from point clouds collected from the terrestrial, mobile, and aerial stances.

In partnership with the USGS National UAS Project Office, Andrew joined the UFUAS Research Program in 2016 to assist in the collection and processing of lidar, photogrammetric, and hyperspectral data.

Andrew served as a teaching assistant at UF from 2013-2017 for the courses of Advanced Photogrammetry, Forest Resource Information Systems, Spatial Measurement Systems, and Geomatics. Prior to that, he worked as a survey technician in Panama City, Florida, in both the public and private sectors.