JOURNAL OF UNMANNED AERIAL SYSTEMS

Online at www.uasjournal.org

Dec. 2017 Volume 3, Issue 1 ISSN 2378-0525

SPONSORED BY

National UAS
Training and Certification Center

uas.sinclair.edu

SPONSORED BY

National UAS
Training and Certification Center

uas.sinclair.edu

TABLE OF CONTENTS

About Our Sponsor 4

From the Editor.. 5

Peer-Reviewed Article
A Comparative Analysis of UAS Crewmember Collegiate Curricula.................................. 6

Meta-analysis of Color Conspicuity for Small Unmanned Aircraft.....................................23

Cyber Attacks and Defense Framework
for Unmanned Aerial Systems (UAS) Environment 37

Operational Organization of Small Unmanned Aerial System Physical Airspace..............59

Assessment of Structure from Motion (SfM) Processing Parameters
on Processing Time, Spatial Accuracy, and Geometric Quality
of Unmanned Aerial System Derived Mapping Products 73

Qualitative Comparison of the Slanted Tailplane Configuration
in the Indonesian BPPT's General Purposes UAV... 100

JOURNAL OF UNMANNED AERIAL SYSTEMS

ABOUT OUR SPONSOR

Sinclair College National UAS Training and Certification Center

Sinclair College's National UAS Training and Certification Center, located in Dayton, Ohio, represents the culmination of a focused vision dedicated to creating one of the most comprehensive and pioneering facilities for the advancement of UAS training and applied research support. Since 2008, Sinclair College has been at the forefront of UAS innovation, creating partnerships, developing leading curriculum, and investing significantly to establish a nationally prominent program dedicated to meeting the workforce needs of the growing UAS industry.

Created through total investments of more than $10 million, Sinclair's UAS Center provides students and researchers with the ability to work with new UAS technologies in an immersive and hands-on environment. The facility supports research, development, and training on vehicles and components through advanced unmanned and manned simulation, sensors, avionics, maintenance, advanced manufacturing and rapid prototyping, data analytics, and wind tunnel labs. Additionally, indoor flight training and testing is made possible in both the UAS Indoor Flight Range and the custom built UAS Indoor Flying Pavilion. Sinclair actively operates UAS in the National Airspace System, originally through 13 Certificates of Authorization and Section 333 Exemptions and now also conducting operations under the Part 107 regulations.

Sinclair strives to remain at the cutting-edge of UAS training and applied research support through collaborations with leading UAS organizations in academia, government, and industry. The college is honored to be included as an partner in both the Federal Aviation Administration ASSURE UAS Center of Excellence and National Science Foundation Center for Unmanned Aircraft Systems, maintains active Educational Partnership Agreements focused on UAS with the Air Force Research Laboratory and the Air Force Institute of Technology, has partnered with NASA through a Space Act Agreement, and supports multiple government and industry commissioned UAS applied research efforts. Sinclair also founded and continues to sponsor the Journal of Unmanned Aerial Systems, a peer-reviewed publication that serves the public as an open-access online resource enabling the development and distribution of knowledge for the UAS industry.

National UAS
Training and Certification Center

EDITORIAL

From the Managing Editor: Dr. Andrew Shepherd

Welcome to the third edition of the Journal of Unmanned Aerial Systems. We are pleased to continue to serve as a leading peer-reviewed and open-source UAS resource for those in academia, government, and industry seeking to contribute, share, and learn from each other. Since our last publication, much has changed and improved related to UAS applications and technologies, progress facilitated in some measure by the authors whose work was selected for inclusion in this volume.

As always, we are grateful to the dedicated volunteers supporting the Journal as Reviewers, Editorial Board, and Publishing Board members. Without their efforts and high standards this publication would not exist. Additionally, I would like to acknowledge the contributions of our included authors. Their responsiveness, patience, and provision of quality content is a testament to their professionalism.

The publication of Volume 3, Issue 1 continues to advance the mission of the Journal of Unmanned Aerial Systems to provide a premier interdisciplinary forum for scholarly dialogue of original research and other salient contributions offered by authors from around the world. We are proud of the impact that the Journal continues to make and are excited as we begin preparations for future publications.

Andrew D. Shepherd

Andrew D. Shepherd, PhD – Managing Editor
Executive Director and Chief Scientist, Unmanned Aerial Systems
Sinclair Community College

JOURNAL OF UNMANNED AERIAL SYSTEMS

PEER-REVIEWED ARTICLE

A COMPARATIVE ANALYSIS OF UAS CREWMEMBER COLLEGIATE CURRICULA

Andrew Wentworth
UAS Pilot and Researcher, Alaska Center for Unmanned Aircraft Systems Integration (ACUASI) at the University of Alaska Fairbanks

ABSTRACT

This research was a comparative analysis of undergraduate degree-granting collegiate curricula for unmanned aircraft system (UAS) crewmembers. To keep up with the civil and public-use UAS industry demand for competent unmanned aircraft crewmembers, collegiate curricula are being developed at a rapid pace. However, the absence of Federal Aviation Regulations for certification requirements of crewmembers of any UAS greater than 55 pounds, leads to concerns regarding the standardization of unmanned aviation crewmember curricula. Curricula are comprised of educational goals, educational experiences to meet those goals, how these educational experiences are organized, and how these goals are verified. This research focused on a comparative analysis of what academic topics are taught at 18 colleges offering Bachelor or Associate of Science degrees. All but four of the colleges required some sort of manned pilot certification, all offered hands-on training with sUAS or a simulator; however, two did not have any UAS-specific academic topics. Overall the largest relative variation was found in the UAS-specific topics, as measured by the coefficient of variation between topic-required credit hours. This variation raises concerns regarding student employability, matriculation, and workforce stability. Further research is recommended after the Federal Aviation Administration promulgates regulations for >55-pound UAS crewmember certification, using a larger sample set of colleges with more detailed course content descriptions.

Keywords: unmanned aviation, crewmember, education, standardization, collegiate curriculum

Key Terms (Definitions)

Crewmember	The FAA defines UAS crewmembers as pilots, sensor/payload operators, visual observers, and any people required for safe flight operations (Federal Aviation Administration, 2013).
Curriculum	Curriculum is comprised of four parts: (a) educational goals; (b) educational experiences to meet those goals, (c) organization of educational experiences; (d) and verification of elements to meet initially identified goals (Tyler, 1949). Curriculum is the singular form of curricula.
Education	Education is the cognitive capabilities garnered from learning theory and is primarily focused on understanding concepts (Cross, 1996). In short, it contributes to what you know.
Public Aircraft	In accordance with 49 U.S.C. 140102(a)(41) and 14CFR 1.1, public use aircraft are those performing noncommercial governmental functions such as national defense, intelligence missions, firefighting, search-and-rescue, law enforcement, aeronautical research, or biological or geological resource management.
Small Unmanned Aircraft	The term ‘‘small unmanned aircraft’’ means an unmanned aircraft weighing less than 55 pounds (112th Congress, 2012).
Training	“Training is measured by what you can do when you’ve completed it” and is primarily focused on attaining skills (Cross, 1996).
Unmanned Aircraft	An unmanned aircraft is any aircraft without the possibility of human intervention from onboard the vehicle (Federal Aviation Administration, 2015b).
Unmanned Aircraft System	An unmanned aircraft system is the unmanned aircraft and all associated equipment necessary to operate the system (Federal Aviation Administration, 2015c).

Introduction

By 2025, the projected economic growth in unmanned aviation will exceed $82.1 billion and result in over 100,000 new jobs being created, many of which will require unique education and certification (Association for Unmanned Vehicle Systems International, 2013). This requirement has resulted in a rapid effort to educate and train unmanned aircraft system (UAS) crewmembers to meet the expected demand. Undergraduate collegiate institutions that first establish academic dominance in the unmanned aviation crewmember field could derive significant economic benefit and become a boon to their local UAS economies (Barber, 2014).

In pursuit of these benefits, colleges across the country are developing curricula to provide Associate and Bachelor of Science degrees in unmanned aviation fields, like offerings from colleges that have provided hands-on aviation training and the accompanying degrees in aeronautics (Siebenmark, 2015). However, failure to provide UAS crewmember graduates that are accepted and recognized by the unmanned aircraft industry could have a negative impact on the continued economic health of these colleges. This research will focus on a comparative analysis of the unmanned aviation curricula of various institutions with the objective to characterize the level of standardization of unmanned aviation curricula and recommend areas for potential improvement.

Literature Review

Terwilliger (2013) characterized the growing UAS industry as a three-legged stool comprised of government, industry, and academia. Four aspects of academic involvement are outlined: (a) research and development (R&D); (b) public outreach; (c) addressing the disparity between the UAS fleet and infrastructure; and (d) identifying needs of the UAS community (Terwilliger, 2013). The most relevant need identified, in relation to this research, is for training and education opportunities to prepare the workforce and to provide professional development options. Academia is a stakeholder in the growth of UAS, and as such has much to either gain or lose depending on how the growth is managed.

Perrit and Sprague (2015) engaged in research regarding the ramifications of civil and public use UAS entitled “Drones”. Of specific interest was their analysis of meeting UAS workforce demands. They concluded that UAS crewmember curricula appear to be outgrowths of established aviation curricula, whereby students attain FAA aircraft crewmember certifications, but also include courses specifically targeted at the UAS field (Perrit & Sprague, 2015).

Methodology

Quantitative data were gathered in the form of total credit hours required, the distribution of topics associated with the credit hours, and the proportion of hands-on or laboratory training credit hours. These data were obtained from publicly-available college course catalogs. Descriptive statistics of the credit counts for each topic were created with the online statistical package StatCrunch (Integrated Analytics LLC, 2015). The descriptive statistics were: (a) mean; (b) median; (c) standard deviation; and (d) coefficient of variation (CV).

Data obtained from the curricula were grouped into five areas: (a) overall credits required; (b) general education credits required; (c) aviation credits required; (d) unmanned aviation-centric credits required; and (e) hands-on training credits required. Aviation credits required were determined by the number of credits that have aviation-related topics in the course titles and course descriptions, and unmanned aviation-centric credits were determined from the classes that have UAS-related topics in the course titles and descriptions. Hands-on training credits were determined by examining the classes that have UAS lab or field work requirements. All other required credits were considered general education. All categories were mutually exclusive. Bachelor and Associate degrees were analyzed separately.

Unmanned Aircraft System Crewmember Associate of Applied Science Degrees Green River Community College

Program Summary. Green River Community College is a two-year undergraduate college in Auburn, Washington, and has an established manned aviation department (Green River College, 2015a). From this foundation, an Associate in Applied Science in UAS degree has been created (Green River College, 2015b). The curriculum requires that students attain their private pilot *Airplane Single Engine, Land* (ASEL) certificate.

Curriculum Distribution. The distribution of the curriculum between the four parts outlined in this research is as follows: (a) 35 quarter hours of general education credits, (b) 40 quarter hours of aviation credits, (c) 10 quarter hours of unmanned-centric credits, (d) five quarter hours of UAS hands-on or laboratory credits, and (e) 90 quarter hours total credits to complete the degree (Green River College, 2015b). Division of the academic calendar into quarters as opposed to semesters sets this institution apart from other colleges; hence a correction factor of 1.5 (The Best Schools, 2017) has been applied to the credits in Table 1.

Central Oregon Community College

Program Summary. The Associate of Applied Science in Aviation – UAS Operations from Central Oregon Community College (COCC) is an outgrowth of their manned aviation curriculum (Orcelletto, 2015). Students are required to obtain their FAA private pilot, instrument, and commercial ratings in manned airplanes (Central Oregon Community College, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 47 quarter hours of general education credits; (b) 32 quarter hours of aviation credits, (c) 13 quarter hours of unmanned-centric credits, (d) eight quarter hours of UAS hands-on or laboratory credits, and (e) 100 quarter hours total credits to complete the degree (Central Oregon Community College, 2015). Division of the academic calendar into quarters as opposed to semesters sets this institution apart from other colleges; hence a correction factor of 1.5 (The Best Schools, 2017) was applied to the credits in Table 1.

Community College of Beaver County

Program Summary. The Community College of Beaver County (CCBC) in western Pennsylvania has an established manned aviation curriculum with their UAS curriculum as a byproduct (Community College of Beaver County, 2015). Their unmanned aircraft degree is called Associate of Applied Science in Unmanned Aerial Vehicle (UAV).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 22 semester hours of general education credits, (b) 31 semester hours of aviation credits, (c) zero semester hours of unmanned-centric credits, (d) 12 semester hours of UAS hands-on or laboratory credits, and (e) 65 semester hours total credits to complete the degree (Community College of Beaver County, 2015). The four courses that comprise the hands-on portion of the curriculum consist of three flying classes and a class where the students build an unmanned aircraft (U. Matuszak, personal communication, December 9, 2015).

Northwestern Michigan College

Program Summary. The UAS degree at Northwestern Michigan College (NMC) is entitled Associate of Science in Aviation, but they have a UAS core of classes and hands-on training that can be selected as electives within the curriculum (Northwestern Michigan College, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows (a) 18 semester hours of general education credits, (b) 37 semester hours of aviation credits, (c) zero semester hours of unmanned-centric credits, (d) 10 semester hours of UAS hands-on or laboratory credits, and (e) 65 semester hours total credits to complete the degree (Northwestern Michigan College, 2015).

Cochise College

Program Summary. The UAS crewmember degree offering at Cochise College is the Associate of Applied Science (AAS) in UAS. Cochise College offers a manned aviation curriculum that is FAA Part 141 certified, but the UAS degree is not an outgrowth of manned aircraft curricula, but rather from the college's collaborative relationship with U.S. Army UAS programs at Fort Huachuca, Arizona (Cochise College, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 23 semester hours of general education credits, (b) 27 semester hours of aviation credits, (c) six semester hours of unmanned-centric credits, (d) eight semester hours of UAS hands-on or laboratory credits, and (e) 64 semester hours total credits to complete the degree (Cochise College, 2015).

Sinclair Community College

Program Summary. The Sinclair Community College Associate of Applied Science in Unmanned Aerial Systems program is an outgrowth of the college's short term technical certificates which are, in turn, outgrowths of the college's established manned aviation curriculum (A. Shephard, personal communication, November 4, 2013). The process of starting with technical certificates and developing the curriculum to enable offering an Associate degree began in 2008 (Lambert, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 32 semester hours of general education credits, (b) 13 semester hours of aviation credits, (c) 14 semester hours of unmanned-centric credits, (d) three semester hours of UAS hands-on or laboratory credits, and (e) 62 semester hours of total credits to complete the degree (Sinclair UAS Training and Certification Center, 2015).

Hinds Community College

Program Summary. The UAS degree offering at Hinds Community College (HCC) is the Associate of Applied Science in Aviation Technology with a focus in UAS Technology (Hinds Community College, 2015a). The degree offering is three years old and is an outgrowth of the college's aviation maintenance and FAA Part 141 schools (D. Lott, personal communication, December 19, 2015). The students are not required to obtain their private pilot certification within the curriculum, but are encouraged to do so on their own or as part of their electives (D. Lott, personal communication, December 19, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 18 semester hours of general education credits, (b) 12 semester hours of aviation credits, (c) three semester hours of unmanned-centric credits, (d) 27 semester hours of UAS hands-on or laboratory credits, and (e) 60 semester hours of total credits to complete the degree. This distribution was determined by including the recommended private pilot courses into the elective positions.

Yavapai College

Program Summary. Yavapai College is in Prescott, Arizona and their UAS degree offering is called an Associate of Applied Science in Aviation Technology with a concentration in UAS (Yavapai College, 2015). The curriculum does not require the students to attain their manned airman certifications (Yavapai College, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 31 semester hours of general education credits, (b) three semester hours of aviation credits, (c) 18 semester hours of unmanned-centric credits, (d) eight semester hours of UAS hands-on or laboratory credits, and (e) 60 semester hours of total credits to complete the degree (Yavapai College, 2015).

Summary of Unmanned Aircraft System Crewmember Associate Degrees

Examination of multiple collegiate course catalogs and public information paired with phone calls and e-mails revealed how much of each curriculum was distributed between academic UAS material and hands-on material. Additionally, some colleges lacked any purely academic UAS curriculum and provided all their UAS related material in the hands-on regime, revealing a methodological limitation of making the two categories mutually exclusive. Two examples of this approach can be seen in Table 1 (CCBC and NMC). However, every college had some form of hands-on training.

Table 1
Credit Hours by College Institution and Topic for Associate Degrees

College Name	Abbrev.	General Education Credit hours	Aviation Education Credit hours	Unmanned Education Credit hours	Hands-on or lab credit hours	Total Credits to complete degree
Green River College	GRC	23.33(35)	26.66(40)	6.66(10)	3.33(5)	60(90)
Central Oregon Community College	COCC	31.33(47)	21.33(32)	8.66(13)	5.33(8)	66.66(100)
Community College of Beaver County	CCBC	22	31	0	12	65
Northwestern Michigan College	NMC	18	37	0	10	65
Cochise College		23	27	6	8	64
Sinclair Community College	SCC	32	13	14	3	62
Hinds Community College	HCC	18	12	3	27	60
Yavapai College		31	3	18	8	60

Note: ( ) parentheses indicate quarter credit hours

There was notable variation in the amount (e.g. private, commercial, instrument) of FAA manned pilot certificates required as part of the curriculum. Table 2 indicates that SCC, HCC, and Yavapai College did not have any required manned pilot certifications while Central Oregon Community College was the most demanding in this respect, requiring a commercial certificate. Although the FAA Part 107 (2016) regulations for sUAS do not require a manned pilot certificate, the job opportunities of graduates may be restricted for certain employers, such as the Customs and Border Protection which requires that their UAS pilots have a manned pilot commercial certificate (Border Patrol Edu, 2016). Furthermore, graduates without manned pilot certificates will likely be constrained to employment in the small UAS community.

Table 2
FAA Manned Certificates and Ratings Included in Associate Degree Curricula

College Name	Abbrev.	Private	Instrument	Commercial	Multi-Engine	Certified Flight Instructor
Green River College	GRC	Required	Optional			
Central Oregon Community College	COCC	Required	Required	Required		
Community College of Beaver County	CCBC	Required	Required			
Northwestern Michigan College	NMC	Required	Required	Optional	Optional	Optional
Cochise College		Required	Required			
Sinclair Community College	SCC					
Hinds Community College	HCC	Optional				
Yavapai College						

Note: Airplane or Helicopter rating not specified

Statistical Analysis of Unmanned Aircraft System Crewmember Associate Degrees

General education credit hours from Table 1 varied from 18 to 32 with a mean of 24.83 and a standard deviation of 5.84 (Table 3). Aviation credit hours from Table 1 varied from three to 37 with a mean of 21.37 and a standard deviation of 11.29, which was the highest of all topic areas (Table 3). Unmanned academic credit hours from Table 1 varied from zero to 18 credits hours with a mean of 7.04 and a standard deviation of 6.41 (Table 3). Unmanned hands-on training credit hours from Table 1 varied from three to 27 with a mean of 9.58 with a standard deviation of 7.70 (Table 3). Overall credits required for degree completion from Table 1 varied from 60 to 67 (100 quarter hours) with a mean of 62.83 and a standard deviation of 2.68 (Table 3).

Standard deviations are a good measure of how much variation there is in each category, but a ratio called the coefficient of variation (CV) allows relative levels of variance to be compared between categories enabling comparisons between Associate degrees and Bachelor degrees as well as their constituent topics. Lovie (2005) states that the CV is a measure of relative variability, independent of both the units of measurement and the magnitude of the data and is defined as the standard deviation (SD) divided by the mean.

Table 3 indicates that the area of the curriculum with the most variation is UAS education. UAS education's coefficient of .91 approaches 1.0 indicating a low level of standardization of the classroom education in UAS topics. The next most varied category was UAS hands-on training. These two topics (i.e. UAS education and UAS hands-on training) that comprise the UAS specific aspect of the curriculum vary more than the other topics and do raise concerns about the level of overall standardization from college to college.

Table 3
Summary Descriptive Statistics for Topics within UAS Associate Degrees

Curricula Topics	**Mean**	**Median**	**Std. dev.**	**Coefficient of Variation**
General Education Credit Hours	24.83	23.17	5.84	0.24
Aviation Education Credit Hours	21.37	24.00	11.29	0.53
UAS Education Credit Hours	7.04	6.33	6.41	0.91
UAS Training Credit Hours	9.58	8	7.70	0.80
Total Credits to Complete Degree	62.83	63	2.68	0.04

Unmanned Aircraft System Crewmember Bachelor Degrees

University of Louisiana at Monroe

Program Summary. The UAS crewmember degree offering at the University of Louisiana at Monroe (ULM) is called a Bachelor of Science in Aviation with a concentration in UAS (University of Louisiana at Monroe, 2015a). It is a development from the established manned aviation curriculum at ULM and the concentration in UAS was first offered in 2013 (University of Louisiana at Monroe, 2015b).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 78 semester hours of general education credits, (b) 30 semester hours of aviation credits, (c) nine semester hours of unmanned-centric credits, (d) three semester hours of UAS hands-on or laboratory credits, and (e) 120 semester hours of total credits to complete the degree (University of Louisiana at Monroe, 2015a).

Liberty University

Program Summary. The Bachelor of Science in Aeronautics with a concentration in Unmanned Aerial Systems is the most recent addition to their course offerings from the college of aeronautics, in which the students will attain both their private pilot certification and their instrument rating (Liberty University, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 60 semester hours of general education credits, (b) 39 semester hours of aviation credits, (c) six semester hours of unmanned-centric credits, (d) nine semester hours of UAS hands-on or laboratory credits, and (e) 120 semester hours of total credits to complete the degree (Liberty University, 2015).

Embry-Riddle Aeronautical University

Program Summary. Embry-Riddle Aeronautical University (ERAU) offers a Bachelor of Science in Unmanned Aircraft Systems Science degree at both of their residential campuses in Daytona Beach, Florida and Prescott, Arizona (Embry-Riddle Aeronautical University, 2015). There are two tracks within the degree: (a) Professional UAS Pilot, and (b) UAS Operations, with the difference being that instead of the FAA manned pilot certification courses as in the professional UAS Pilot track, the UAS Operations track has aviation specific electives (Embry-Riddle Aeronautical University, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 59 semester hours of general education credits, (b) 36 semester hours of aviation credits, (c) 18 semester hours of unmanned-centric credits, (d) nine semester hours of UAS hands-on or laboratory credits, and (e) 122 semester hours of total credits to complete the degree (Embry-Riddle Aeronautical University, 2015).

Kansas State University

Program Summary. Kansas State University's (KSU) UAS degree offering is the Bachelor of Science in Unmanned Aircraft Systems (Kansas State University, 2015a). The UAS department partners with the manned aviation department and the engineering department to leverage the UAS curriculum across a greater portion of the student body (M. Most, personal communication, December 17, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 72 semester hours of general education credits, (b) 40 semester hours of aviation credits, (c) nine semester hours of unmanned-centric credits, (d) six semester hours of UAS hands-on or laboratory credits, and (e) 127 semester hours of total credits to complete the degree (Kansas State University, 2015a).

University of North Dakota

Program Summary. The University of North Dakota was the first collegiate institution to offer a Bachelor's degree in unmanned aircraft (Defense and Aerospace Week, 2014). Their degree is called a Bachelor of Science in Aeronautics with a major in Unmanned Aircraft Systems Operation (University of North Dakota, 2015). North Dakota's designation as a FAA UAS Test Site along with their membership in ASSURE contribute to the strength of the UAS industry in the state (Federal Aviation Administration, 2014)(ASSURE, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 69 semester hours of general education credits, (b) 37 semester hours of aviation credits, (c) 15 semester hours of unmanned-centric credits, (d) four semester hours of UAS hands-on or laboratory credits, and (e) 125 semester hours of total credits to complete the degree (University of North Dakota, 2015).

Indiana State University

Program Summary. Indiana State University's (ISU) UAS degree offering is the Bachelor of Science in Aviation Technology with a major in Unmanned Systems (Indiana State University, 2015a). The degree does not require that students attain any manned pilot certifications, but they are encouraged to do so with their electives (Indiana State University, 2015a). The degree is part of the aviation technology department which is an established part of the college and contributes a large portion to the Unmanned Systems curriculum (Indiana State University, 2015a).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 62 semester hours of general education credits, (b) 40 semester hours of aviation credits, (c) 15 semester hours of unmanned-centric credits, (d) three semester hours of UAS hands-on or laboratory credits, and (e) 120 semester hours of total credits to complete the degree (Indiana State University, 2015b). This distribution was determined by selecting manned aircraft certifications as electives to include private pilot and instrument ratings.

Lewis University

Program Summary. The Lewis University UAS crewmember degree offering is the Bachelor of Science in Unmanned Aircraft Systems from the Aviation and Transportation Department (Lewis University, 2015a). Lewis University also offers a minor in UAS for any of their other degrees due to the broad relevance of the technology, which has the effect of increasing UAS class sizes (Lewis University, 2015b).

Curriculum Distribution. The distribution of the curriculum between the four parts outlined in this research is as follows: (a) 66 semester hours of general education credits, (b) 47 semester hours of aviation credits, (c) nine semester hours of unmanned-centric credits, (d) six semester hours of UAS hands-on or laboratory credits, and (e) 128 semester hours of total credits to complete the degree (Indiana State University, 2015b).

Middle Tennessee State University

Program Summary. Middle Tennessee State University (MTSU) is in Murfreesboro, Tennessee and brands their UAS crewmember degree offering a Bachelor of Science in Aerospace with a concentration in Unmanned Aircraft Systems (Middle Tennessee State University, 2015). Students are required to obtain their FAA Private Pilot certificate as part of their studies (Middle Tennessee State University, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 75 semester hours of general education credits, (b) 30 semester hours of aviation credits, (c) 12 semester hours of unmanned-centric credits, (d) three semester hours of UAS hands-on or laboratory credits, and (e) 120 semester hours of total credits to complete the degree (Middle Tennessee State University, 2015).

LeTourneau University

Program Summary. LeTourneau University is in Longview, Texas and has a UAS degree offering called the Bachelor of Science in Remotely Piloted Aircraft Systems with the pilot concentration (LeTourneau University, 2015). The pilot concentration requires a FAA manned pilot certificate (LeTourneau University, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows: (a) 46 semester hours of general education credits, (b) 58 semester hours of aviation credits, (c) 17 semester hours of unmanned-centric credits, (d) five semester hours of UAS hands-on or laboratory credits, and (e) 126 semester hours of total credits to complete the degree (Liberty University, 2015).

Purdue University

Program Summary. Purdue University in West Lafayette, Indiana has the state's second UAS crewmember undergraduate college offering a Bachelor of Science in Unmanned Aerial Systems through the School of Aviation and Transportation Technology (Purdue Polytechnic, 2015).

Curriculum Distribution. The distribution of the curriculum is as follows (a) 62 semester hours of general education credits, (b) 24 semester hours of aviation credits, (c) 16 semester hours of unmanned-centric credits, (d) 18 semester hours of UAS hands-on or laboratory credits, and (e) 120 semester hours of total credits to complete the degree (Purdue Polytechnic, 2015).

Summary of Unmanned Aircraft System Crewmember Bachelor Degrees

Examination of multiple collegiate institutions course catalogs and public information paired with personal communications revealed the curriculum distribution between academic UAS material and hands-on material. Some colleges combined their hands-on training and classroom instruction and, in doing so, revealed a methodological limitation of making the two categories mutually exclusive for a comparative analysis. However; every college had some form of hands-on training with one exception that only used simulators for hands-on training. Table 4 summarizes the curriculum distributions of the 10 UAS crewmember Bachelor degree granting collegiate institutions.

Table 4
Credit Hours by College Institution and Topic for Bachelor Degrees

College Name	Abbrev.	General Education Credit hours	Aviation Education Credit hours	Unmanned Education Credit hours	Hands-on or lab credit hours	Total Credits to complete degree
University of Louisiana at Monroe	ULM	78	30	9	3	120
Liberty University		60	39	6	9	120
Embry-Riddle Aeronautical University	ERAU	59	36	18	9	122
Kansas State University	KSU	72	40	9	6	127
University of North Dakota	UND	69	37	15	4	125
Indiana State University	ISU	62	40	15	3	120
Lewis U niversity		66	47	9	6	128
Middle Tennessee State University	MTSU	75	30	12	3	120
LeTourneau University		46	58	17	5	126
Purdue University		62	24	16	18	120

There is less variation in the amount of FAA manned Certificates and ratings required in Bachelor degrees than in Associate degrees. Table 5 indicated that ISU had the least required manned pilot certifications while UND had the most, and that the combination of Private and Instrument were the most common.

Table 5

FAA Manned Certificates and Ratings Included in Bachelor Degree Curricula

College Name	Abbrev.	Private	Instrument	Commercial	Multi-Engine	Certified Flight Instructor
University of Louisiana at Monroe	ULM	Required	Required			
Liberty University		Required	Required			
Embry-Riddle Aeronautical University	ERAU	Required	Required	Required		
Kansas State University	KSU	Required	Required			
University of North Dakota	UND	Required	Required	Required	Required	
Indiana State University	ISU	Optional	Optional	Optional		
Lewis University		Required	Required			
Middle Tennessee State University	MTSU	Required				
LeTourneau University		Required	Required			
Purdue University		Required	Required			

Note: Airplane or Helicopter rating not specified

Statistical Analysis of Unmanned Aircraft System Crewmember Bachelor Degrees

General education credit hours from Table 4 varied from 46 to 78 with a mean of 64.9 and a standard deviation of 9.28 (Table 6). Aviation credit hours from Table 4 varied from 24 to 58 with a mean of 38.1 and a standard deviation of 9.54, which was the highest of all topic areas (Table 6). Unmanned academic credit hours from Table 4 varied from six to 18 credits hours with a mean of 12.6 and a standard deviation of 4.14 (Table 6). Unmanned hands-on training credit hours from Table 4 varied from three to 18 with a mean of 6.6 with a standard deviation of 4.60 (Table 6). Overall credits required for degree completion from Table 4 varied from 120 to 128 with a mean of 122.8 and a standard deviation of 3.33 (Table 6).

Table 6 indicates that the area of the curriculum with the most variance is UAS hands-on training. UAS hands-on training's coefficient of .70 for BS degrees indicated that the standard deviation is 70% of the mean, which was better than the UAS hands-on training CV of 0.80 for AS degrees. The next most varied category was UAS education credit hours with a CV of 0.33, which was far better than the UAS education CV of 0.91 for AS degrees. The two topics (i.e. UAS training and UAS education) that make up the UAS specific aspect of the curriculum vary more than any other topics and raise concerns about the level of overall standardization from college to college.

Table 6
Summary Descriptive Statistics for Topics within UAS Bachelor Degrees

Curricula Topics	Mean	Median	Std. dev.	Coefficient of Variation
General Education Credit Hours	64.9	64	9.28	0.14
Aviation Education Credit Hours	38.1	38	9.54	0.25
UAS Education Credit Hours	12.6	13.5	4.14	0.33
UAS Training Credit Hours	6.6	5.5	4.60	0.70
Total Credits to Complete Degree	122.8	121	3.33	0.03

Summary of Findings

Many of these curricula were derived from established manned aviation curricula with the UAS departments emerging as subsets or partners of the aviation departments to enable curriculum sharing and prevent duplication of efforts regarding instructional design. The UAS degrees are striving to meet the needs of their respective regions for competent UAS crewmembers to make economic gains. They are also doing so to draw potential economic gains away from their neighboring states. Some colleges distinguished themselves by their partnerships with local military operations and/or established civil UAS manufacturers that support their UAS programs by providing access to restricted or special use airspace and as subcontractors for UAS flight training.

Many of the sources in this research were news articles from local news outlets that are in the communities where the colleges reside. This is a manifestation of the colleges fulfilling Terwilliger's (2013) recommendation that academia engages in public outreach and education to improve social acceptance of UAS. It is not entirely a selfless act though, the colleges have an interest in drawing students and to do so, they must market their degree offerings as good choices for potential students; transforming the dialog about UAS to one about economic opportunity and revitalization.

The most credit hour variation in undergraduate UAS crewmember curricula was in the UAS-related topics, both academic and hands-on. Differences in the coefficients of variation between credit hours across topics were found in both Associate degrees and Bachelor degrees. Overall the Associate degrees had more variation in every topic than Bachelor degrees (Figure 1). The top two Associate degree topics with the most variation was UAS hands-on training and UAS academic education, which was also true for Bachelor degree curricula, although in the opposite order. Although the Bachelor degree curricula required nearly twice as many overall credit hours as the Associate degrees, the latter had a higher mean hands-on training credit hour requirement.

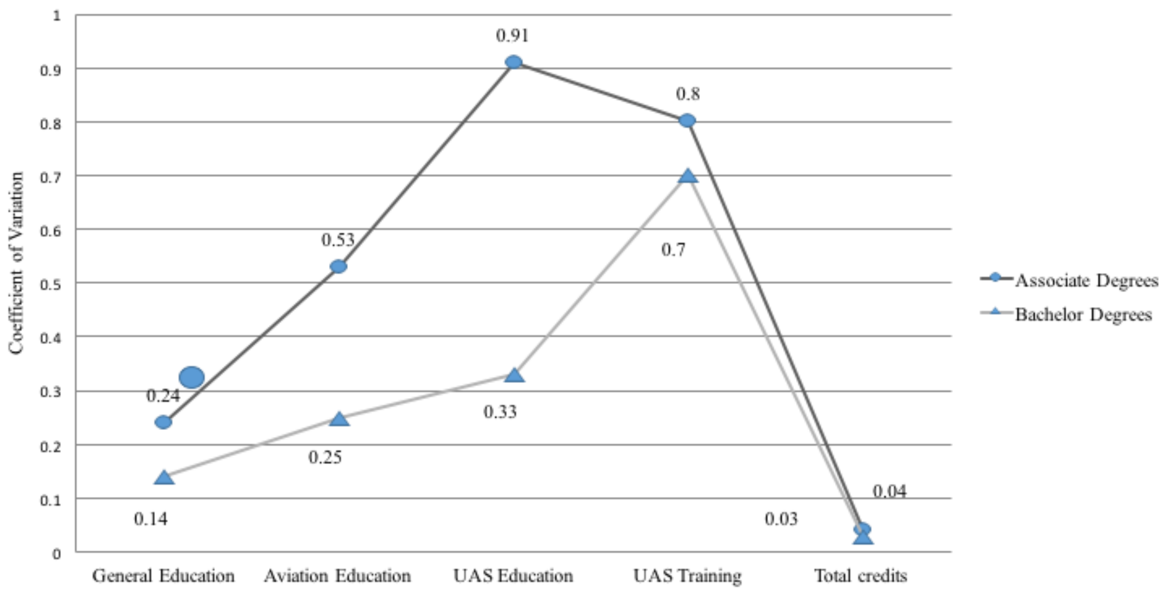


Figure 1. Coefficients of variation by curriculum topic and degree type.

FAA manned pilot certifications that were a part of the curriculum differed from institution to institution, but 78% of institutions included a requirement for at least a private pilot certification in the curriculum. Only one college offering a BS degree made a manned pilot certification optional, while three colleges offering AAS degrees made it either optional or not required.

Conclusions

Overall, BS degrees were more standardized than AAS degrees, with the UAS specific area within the curriculum (i.e. UAS academic topics and UAS hands-on training) showing the least standardization. Specifically; within Associate degrees, the UAS academic material was the least standardized and within Bachelor degrees, the UAS hands-on training was the least standardized. The mean for all the CVs of Associate degrees was 0.50 and the mean for all the CVs of Bachelor degrees was 0.29, illustrating that Bachelor degree curricula were more standardized than Associate degree curricula.

The large relative variability in required credit hours within topics or from degree to degree has ramifications for the growing industry. Graduates from different undergraduate colleges will have different strengths and weaknesses. Specifically, and most critically, graduates of Associate degree curricula are likely to have more variation in their level of UAS academic education than graduates of Bachelor degree curricula, which may result in graduates that do not understand the same breadth or depth of UAS topics. Despite the relative low variability in aviation and overall credit hours for graduates with Bachelor degrees, they still had a large variation in their level of UAS flight experience.

Employer awareness of non-standardized curricula could pose challenges for graduates in their job hunt as well as for colleges recruiting future students. Employers could be negatively impacted by lack of standardization where multiple employees with similar college degree credentials present differing skillsets, complicating human resource labor planning, professional development, and operational success. Additionally, the lack of standardization creates students that have trouble matriculating from Associate UAS degree curricula to Bachelor UAS degree curricula in pursuit of higher education.

Recommendations

Curriculum Standardization

In order in reduce the relative variability and improve the level of standardization within UAS crewmember undergraduate collegiate curricula, several aspects must be addressed; specifically, FAA manned pilot certifications, UAS academic credits, and UAS hands-on experiences. The following three recommendations require some manner of agreement between the involved schools.

FAA manned pilot certifications are one of the elements within the curriculums lacking standardization. To correct this, colleges should include minimum FAA pilot certification requirements for the completion of UAS crewmember curricula. Associate degrees should require a minimum of a sport pilot certificate, and Bachelor degrees should require a minimum of private pilot certificate in either a helicopter or airplane. This requirement would further elevate graduates from their peers with simply a Part 107 certificate. The inclusion of this material in the curriculum would add very little course work for the students' due to ground school already being part of curriculum and schools would be free to exceed this minimum requirement. In the future, when the regulatory environment matures and more types of unmanned FAA certificates exist, this recommendation could easily become outdated and fall to the wayside.

UAS academic topics should be included in the curriculum to provide the students a foundation of system knowledge regarding the different types of UAS as well as environmental, economic, and political aspects. Some of the colleges did not offer any UAS academic topics, while others either provided dedicated UAS academic classes or combined their UAS academic topics with their UAS hands-on training.

UAS hands-on experience should be increased so that curriculum standards for colleges are similar in scope to the aeronautical experience required for existing manned pilot certificates under 14CFR Part 61. Required UAS flight hours, not including simulator time, should have different criteria for Associate degrees and Bachelor degrees so that the graduates' unmanned flight experience exceeds their manned flight experience thus preventing graduates from unmanned centric curricula having hundreds of manned flight hours, but only a handful of unmanned fight hours. These unmanned flight hours should be comprised of a variety of UAS types, but the exact distribution will be determined by the nature of the region that the university is trying to support. These degrees should be viewed as unmanned aviation degrees with a seasoning of manned material rather than manned aviation degrees with an unmanned flavor.

Further Analysis

The goal of UAS crewmember undergraduate college curricula has been to meet the workforce demands of a growing industry, but how well that is being accomplished requires further research. This research only addressed how standardized the current paradigm is. Therefore; a longitudinal study is recommended to measure the work experiences and employability of graduates of these curricula, over the course of their careers. This recommended course of action could also measure how many students pursued higher UAS crewmember education and how many attained additional manned certifications.

Additionally; it would allow the effectiveness of the curricula to be compared against one another and to determine which curricula were superior to the others. This could form the basis of discovering a model curriculum. Further refining this model curriculum would be an industry study examining the qualifications employers' desire in graduates of UAS crewmember curricula. The result could then be used to design recommendations to academic institutions for the desired distribution and inclusion of topics within the curricula.

Recommendation to Improve this Analysis

The relative variability of curricula standardization should be investigated at a finer level by expanding from the top-level course descriptions provided in the college catalogs, down to the content of individual syllabi for each class. This would enhance the reliability of correct categorization of curricula topics as hands-on or academic in nature, and allow blended classes to be more accurately represented in the curriculum distribution. The limitation to this approach is that collegiate institutions may not be willing to share their individual course syllabi due to concerns about curriculum development costs and retaining their intellectual property.

REFERENCES

112th Congress. (2012, February 1). *Public Law 112-95*. Washington: Government Printing Office.

Association for Unmanned Vehicle Systems International. (2013). *The Economic Impact of Unmanned Aircraft Systems Integration in the United States.* Arlington: AUVSI.

ASSURE. (2015, December 20). *The FAA's Center of Excellence for UAS Research*. Retrieved from http://www.assureuas.org/about.php

Barber, B. (2014, October 25). *Sinclair Sees UAS Payoff in Future Jobs*. Retrieved from http://www.daytondailynews.com/news/news/sinclair-sees-uas-payoff-in-future-jobs/nhrSL/

Border Patrol Edu. (2016, January 3). *How to Become a UAV Pilot with Customs and Border Protection*. Retrieved from http://www.borderpatroledu.org/become-uav-pilot/

Central Oregon Community College. (2015, December 19). *Aviation - UAS Operations Program Description*. Retrieved from http://www.cocc.edu/uploadedfiles/academic_programs/aviation/2014-15-av-uas-pilot-program.pdf

Cochise College. (2015, December 19). *Unmanned Aircraft Systems and Operations - Associate of Applied Science (Major Code - UAS)*. Retrieved from http://www.cochise.edu/catalog/unmanned-aircraft-systems-and-operations---associate-of-applied-science-major-code---uas.htm

Community College of Beaver County. (2015, December 19). *Course Catalog Unmanned Aerial Vehicle*. Retrieved from http://www.ccbc.edu/UAV

Cross, J. (1996). *Training vs. Education: A Distinction That Makes A Difference.* San Francisco: Omega Performance.

Defense and Aerospace Week. (2014). North Dakota Department of Commerce; North Dakota Leads UAS Research with First FAA - Mandated Test Site. *Defense & Aerospace Week*, 572.

Embry-Riddle Aeronautical University. (2015, December 19). *Bachelor of Science in Unmanned Aircraft Systems Science*. Retrieved from http://daytonabeach.erau.edu/degrees/bachelor/unmanned-aircraft-systems-science/index.html

Federal Aviation Administration. (2013). *Integration of Civil Unmanned Aircraft Systems (UAS) in the National Airspace System (NAS) Roadmap First Edition*. Washington: Department of Transportation.

Federal Aviation Administration. (2014, April 21). *FAA Announces First UAS Test Site Operational*. Retrieved from https://www.faa.gov/news/press_releases/news_story.cfm?newsId=16154

Federal Aviation Administration. (2015a). *14 Code of Federal Regulations Part 1.1.* Washington: Department of Transportation.

Federal Aviation Administration. (2015b, June 1). *Small UAS Notice of Proposed Rulemaking (NPRM)*. Retrieved from https://www.faa.gov/uas/nprm/

Federal Aviation Administration. (2015c, December 18). *Unmanned Aircraft Systems Frequently Asked Questions*. Retrieved from Unmanned Aircraft Systems: https://www.faa.gov/uas/faq/#qn19

Federal Aviation Administration. (2016). *14 Code of Federal Regulations Part 107.* Washington, D.C.: Department of Transportation.

Green River College. (2015a, December 19). *About us*. Retrieved from http://www.greenriver.edu/about-us.htm

Green River College. (2015b, December 19). *Unmanned Aerial Systems.* Retrieved from http://www.greenriver.edu/Documents/academic-programs/degrees/proftech/aviation_tech_unmanned_aerial_sytems_aas.pdf

Hinds Community College. (2015a, December 19). *Aviation Technology*. Retrieved from http://catalog.hindscc.edu/preview_program.php?catoid=9&poid=728&returnto=226

Indiana State University. (2015a, December 21). *BS Unmanned Systems*. Retrieved from http://www.indstate.edu/academics/undergraduate/majors/unmanned-systems

Indiana State University. (2015b, December 21). *BS UMS Degree Map*. Retrieved from http://www2.indstate.edu/academicaffairs/docs/fouryearplans/201516/Template%20Unmanned%20Systems-Final.pdf

Integrated Analytics LLC. (2015, December 19). *StatCrunch - Data Analysis on the Web*. Retrieved from http://www.statcrunch.com/index.php

Kansas State University. (2015a, July 7). *Kansas State Polytechnic: College of Technology and Aviation - Unmanned Aircraft Systems.* Retrieved from https://polytechnic.k-state.edu/academics/studyguides/UAS.pdf

Lambert, L. (2015, August 24). *Sinclair Unveils Ohio's first one-stop drone facility*. Retrieved from http://www.mydaytondailynews.com/news/news/sinclair-unveils-ohios-first-one-stop-drone-facili/nnQWN/

LeTourneau University. (2015, December 23). *REMOTELY PILOTED AIRCRAFT SYSTEMS, B.S.* Retrieved from http://www.letu.edu/opencms/opencms/_Academics/Aero-Science/degrees/Remotely_Piloted_Aircraft_Systems/index.html

Lewis University. (2015a, December 21). *Unmanned Aircraft Systems / Bachelor of Science.* Retrieved from http://lewisu.smartcatalogiq.com/en/Undergrad-2015-2016/Undergraduate-Catalog/College-of-Arts-and-Sciences/Aviation-Flight-Management/Unmanned-Aircraft-Systems

Lewis University. (2015b, December 21). *Undergraduate Catalog 2015-2016 > College of Arts and Sciences > Aviation Flight Management > Unmanned Aircraft Systems / Minor*. Retrieved from http://lewisu.smartcatalogiq.com/Undergrad-2015-2016/undergraduate-catalog/College-of-Arts-and-Sciences/Aviation-Flight-Management/Unmanned-Aircraft-Systems-Minor

Liberty University. (2015, June 30). *Academic Catalog*. Retrieved from https://www.liberty.edu/index.cfm?PID=31096

Lovie, P. (2005). Coefficient of Variation. In *Encyclopedia of Statistics in Behavoral Science.*

Middle Tennessee State University. (2015, October 16). *Unmanned Aircraft Systems (UAS) Operations*. Retrieved from http://www.mtsu.edu/programs/unmanned-aircraft/#t-3

Northwestern Michigan College. (2015, December 19). *Aviation Program Requirements*. Retrieved from https://www.nmc.edu/programs/academic-programs/aviation/program-requirements.html

Orcelletto, L. (2015, June 4). *COCC's Unmanned Aerial Systems Program Has Taken Flight*. Retrieved from http://www.cocc.edu/announcements/cocc-s-unmanned-aerial-systems-program-has-taken-flight/

Perrit, H., & Sprague, E. (2015). Drones. *Vanderbilt Journal of Entertainment and Technology Law*.

Purdue Polytechnic. (2015, December 23). *Unmanned Aerial Systems*. Retrieved from https://polytechnic.purdue.edu/degrees/unmanned-aerial-systems

Siebenmark, J. (2015, November 23). *K-State in Salina to Train Commercial Drone Pilots*. Retrieved from http://www.kansas.com/news/business/aviation/article46086315.html

Sinclair UAS Training and Certification Center. (2015, December 7). *Sinclair UAS Training and Certification Center Programs*. Retrieved from http://uas.sinclair.edu/programs.html

Terwilliger, B. (2013). How Can Higher Education Best Support UAS Growth in America. *Collegiate Aviation Review*, 1-5.

The Best Schools. (2017). *Quarter credits versus semester credits*. Retrieved from The Best Schools: http://www.thebestschools.org/degrees/quarter-credits-versus-semester-credits/

Tyler, R. W. (1949). *Basic Principles of Curriculum and Instruction.* Chicago: The University of Chicago Press.

University of Louisiana at Monroe. (2015a, December 19). *Unmanned Aircraft Systems Concentration*. Retrieved from University of Louisiana at Monroe Course Catalog: http://catalog.ulm.edu/preview_program.php?catoid=20&poid=2449&returnto=2481

University of Louisiana at Monroe. (2015b, December 19). *Welcome to the Professional Aviation Program at ULM*. Retrieved from College of Business & Social Sciences Aviation: http://ulm.edu/cbss/aviation/

University of North Dakota. (2015, December 20). *Undergraduate Programs and Courses - Aviation*. Retrieved from http://und-public.courseleaf.com/undergraduateacademicinformation/departmentalcoursesprograms/aviation/#degreestext

Yavapai College. (2015, December 19). *Associate of Applied Science - Aviation Technology (Airplane/Helicopter/Ops/UAS)*. Retrieved from http://www.yc.edu/academics/degrees-and-certificates/associate-of-applied-science---aviation-technology/119

JOURNAL OF UNMANNED AERIAL SYSTEMS

PEER-REVIEWED ARTICLE

META-ANALYSIS OF COLOR CONSPICUITY FOR SMALL UNMANNED AIRCRAFT

Darryl A. White
Alumnus, Embry-Riddle Aeronautical University

ABSTRACT

During the development of the Small Unmanned Aircraft Systems rule under Part 107 in Title 14 of the Code of Federal Regulations, the Federal Aviation Administration reported that they do not have any data that would indicate what color, if any, would enhance the conspicuity of small unmanned aircraft. By definition, conspicuity is the quality or state of being conspicuous or how easily it can be seen or noticed. A meta-analysis of archival data determined that fluorescent red-orange color schemes would most effectively increase the conspicuity of small unmanned aircraft to help establish and maintain unaided visual contact (other than corrective lenses and sunglasses) by ground operators/visual observers and pilots of manned aircraft under day visual meteorological conditions. The use of fluorescent paint schemes on airplane and rotorcraft type small unmanned aircraft, irrespective of their size, increased the visibility, detectability, range, and conspicuity. Fluorescent paint schemes were qualitatively ranked first and had a mean detection range of 2.16 miles versus a 1.0 mile range for non-fluorescent enamel paint schemes. Civil Aviation Authorities should consider the use of fluorescent colors in the development of future regulations and requirements.

Keywords: unmanned aircraft, color conspicuity, visual line-of-sight, detection range

Meta-Analysis of Color Conspicuity for Small Unmanned Aircraft

The rapid and escalating number of small unmanned aircraft being introduced into the National Airspace System (NAS), without any equipment providing a transponding capability to electronically indicate their position to other aircraft, has resulted in an increasing number of incidents between civil small unmanned aircraft and manned aircraft. A search of the Federal Aviation Administration (FAA) Near Mid-Air Collision System (NMACS) database, utilizing the keywords, drone and unmanned, revealed 65 pilot reported NMAC's between unmanned aircraft systems (UAS) and manned aircraft from January 2016 to August 2016 (Federal Aviation Administration, 2016). Out of the 65 reported incidents, seven incidents were evaluated and categorized as a critical incident (FAA, 2016). An incident is categorized as a critical when aircraft separation is less than 100 feet and a collision was avoided due to chance, rather than any actions taken by a pilot (Federal Aviation Administration, 2010, p. 5-5). These numbers can only be expected to rise as more unmanned aircraft begin to operate. The FAA also documents sightings from pilots, citizens and law enforcement in the UAS Sightings Report and recorded 1,346 events between November 2014 and January 2016 (FAA, 2016). For example, on January 1, 2016, a pilot reported that a "drone" passed below the aircraft between 10 or 15 feet, but could not give the color or type (FAA, 2016). Not only was the UAS dangerously close to the aircraft but the pilot was unable to identify the color of the system even within such close proximity. An evaluation of color conspicuity could help develop recommendations on the color or schemes that should be employed by UAS to reduce visual acquisition time by pilots of manned aircraft and enhance the small unmanned aircraft (sUA) to maintain visual line-of-sight contact with the platform under operation in order to remain well clear, see (sense)-and-avoid, and reduce collision hazards with manned aircraft.

The emergence of civil unmanned aircraft led to the passage of the Federal Aviation Administration Modernization and Reform Act of 2012. The FAA Modernization and Reform Act established a timeline of events and provided the authorization for developing unmanned aircraft regulations. The Modernization and Reform Act led to the FAA promulgating the Small Unmanned Aircraft Rule 14CFR Part 107 in June of 2016. The Part 107 ruling outlines regulations that apply to the operation of small unmanned aircraft within the National Airspace System (NAS). An unmanned aircraft that weighs less than 55 pounds is defined as a small unmanned aircraft system according to the FAA Modernization and Reform Act, 49 U.S.C 40101 (2012).

Operators of small unmanned aircraft must follow several key provisions of the Part 107 ruling. Unmanned aircraft are restricted to daylight-only operations, a maximum of 400 feet above ground level (AGL), a weather visibility of at least 3 miles, and must remain within visual line-of-sight (VLOS) from the pilot in command (PIC) or alternatively, the visual observer (VO) (Federal Aviation Administration, 2016, p. 42006). Maintaining visual contact between manned and unmanned aircraft and between sUA and the sUA PIC is an important aspect of ensuring safety within the NAS, and to comply with the regulatory requirements under 14CFR §§107.3, 107.31, 107.33, and 107.37 for sUA and 14CFR §§91.111, 91.113 and 91.181 for manned aircraft, which address see (sense) and avoid, remaining well clear of other aircraft, and collision avoidance (Federal Aviation Administration, 2016, p. 42006).

The allowable operational horizontal range of an unmanned system is limited by human vision capabilities because the remote PIC must be able to maintain unaided visual contact (other than corrective lenses and sunglasses) with the sUAS (Federal Aviation Administration, 2016, p. 42006). Human vision capabilities are affected by the external physical properties of the sUA (e.g. paint scheme, external lighting, dimensions) and the environment (e.g. general level of ambient natural and artificial illumination, sun angle, meteorological conditions, transparency of the atmosphere, terrain masking, background contrast).

Concerns over the ability to see small unmanned aircraft have been raised by commenters during the development of the Small Unmanned Aircraft Rule Part 107. Commenters asserted "that small unmanned aircraft may be difficult to see, from both the ground and from other aircraft operating in the NAS" (Federal Aviation Administration, 2016, p. 42114). The inability to easily see or identify unmanned aircraft could be due to the colors or schemes currently utilized in sUAS designs. The Air Line Pilots Association International (ALPA) commented that UAS's may be almost all black or all white and that they can be difficult to see against a non-contrasting background (Federal Aviation Administration, 2016, p. 42114).

Increasing platform conspicuity could enhance a remote PIC's ability to maintain visual line-of-sight. An increase in the ability to maintain VLOS could also result in a greater allowable operational horizontal range. While developing the Part 107 Rule the FAA acknowledged that they do not have any data that supports the use of any color, or colors, to enhance the conspicuity of small unmanned aircraft but previous research into color conspicuity has been beneficial in other fields. Hunter safety improved by increasing the conspicuity of hunting clothing. Researchers discovered that in more than 13,000 sightings observers preferred the use of fluorescent orange paint to increase hunter conspicuity (Federal Aviation Agency, 1961). A study by King and Solomon (1995) explored the influence color has on the number of accidents involving fire vehicles. Data collected from the Dallas Fire Department revealed that the likelihood of visibility related accidents was perhaps three times greater for fire vehicles painted in a red and white color scheme versus a lime-yellow and white color scheme. (King and Solomon, 1995). The lime-yellow and white color scheme resulted in drivers having greater situational awareness. Using the concepts presented in the two previous examples could increase the conspicuity of small unmanned aircraft.

Commenters suggested several options to help increase sUAS conspicuity during the development of the Part 107 ruling. Options suggested to the FAA include the use of alternating sections of aviation orange and white, a paint scheme of a black bottom with a mostly white top and at least two areas painted florescent/aviation orange, alternating aviation orange and red paint, and utilizing bright neon orange, red, or green (Federal Aviation Administration, 2016, p. 42114). The research presented focuses on determining if any colors or patterns could help increase small unmanned aircraft conspicuity.

Literature Review

Research into aircraft conspicuity has been largely focused on manned aircraft, but conspicuity has been studied in a variety of fields and applications. Many of the colors and schemes suggested by commenters during the 14CFR Part 107 rulemaking process have been previously studied.

Rasmussen, Vaughan, and Welsh (1978) studied the conspicuity of propeller and tail rotor paint schemes. Their research sought to identify a color scheme that would enhance personnel safety on the ground in the vicinity of an aircraft by increasing conspicuity. The study involved 30 volunteers rating three paint schemes for propellers and another two schemes for tail rotor blades. The color schemes employed for the study include; (a) asymmetrical black and white stripes; (b) yellow painted tip; (c) white painted tip with a red stripe; and (d) symmetrical stripes using red, black, and white. Table 1 depicts the mean rank order by viewing angle for the three propeller color schemes studied while Table 2 represents the distribution of the two tail rotor schemes.

Table 1

Mean rank order of propeller conspicuity by paint scheme

	Paint Scheme		
Viewing Angle	Black and White	Red and White	Yellow
Upward	1.53	1.80	2.67
Eye Level	1.27	2.0	2.73
Downward	1.27	1.93	2.80
Combined	1.36	1.91	2.73

Note. Adapted from “Conspicuity assessment of selected propeller and tail rotor paint schemes” by K.W. Welsh, J.A. Vaughan, and P.G. Rasmussen, 1978, No. FAA-AM-78-29, Federal Aviation Administration Washington DC Office of Aviation Medicine.

Table 2
Mean rank order of tail rotor conspicuity by paint scheme

		Paint Scheme	
Viewing Angle	Subjective Ranking	Black and White	Red and White
Upward	Most	26	4
Eye Level	Most	26	4
Downward	Most	29	1
Combined	Most	81	9

Note. Adapted from "Conspicuity assessment of selected propeller and tail rotor paint schemes" by K.W. Welsh, J.A. Vaughan, and P.G. Rasmussen, 1978, No. FAA-AM-78-29, Federal Aviation Administration Washington DC Office of Aviation Medicine.

Volunteers selected or ranked the black and white stripes scheme as most conspicuous for both propellers and tail rotors. The differences in Table 1 paints schemes were significant at the .001 level after researchers performed a two-way analysis of variance for ranked data. Rasmussen, Vaughn, and Welsh (1978) also performed a Chi-square analysis on the Table 2 tail rotor paints scheme. The perceived differences between paint schemes were determined to be statistically significant ($p < .01$). During the development of 14CFR Part 107 commenters suggested that alternating black and white paint could increase platform conspicuity.

The Applied Psychology Corporation (1961) prepared several reports on the role of paint in mid-air collision prevention for the Federal Aviation Agency. The reports focused on the use of exterior surface treatments to aid in visual collision avoidance. The report found firstly that some paint on the aircraft is better than the unpainted metal surface due to visual blending. Their research further revealed that an optimal paint scheme utilized positive and negative brightness contrast and color contrast. The report particularly recommended using dark colors on the bottom of the fuselage and wings while also using white or any bright color on the top. A color scheme suggested to the FAA used a black bottom, a mostly white top, and included at least two areas of fluorescent orange. The Applied Psychology Corporation also concluded that fluorescent orange and red paints are preferred over other fluorescent colors and enamels of all colors. Identification ranges for fluorescent red-orange, orange, and yellow-orange averaged 2.3 miles versus a 1.0-mile range average for non-fluorescent reds and oranges or other enamels of other color. Figure 1 provides a visual depiction of the average distances for each color scheme tested.

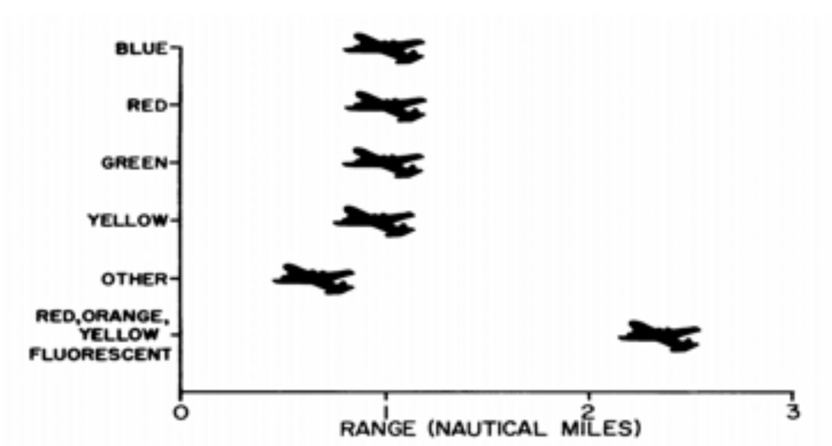


Figure 1. Average distance at which aircraft predominate color was identified. Adapted from "The Role of Paint in Mid-Air Collision Prevention" by the Federal Aviation Agency, 1961, Applied Psychology Corporation AD No. 273691.

The United States Air Force initiated a program to decrease mid-air collisions as reported by Baker (1960). The Air Force utilized fluorescent orange paint on the wing tips while also painting wide bands around the nose and aft sections of the fuselage. The Air Forces Air Training Command noted a consistent decline in mid-air collisions that closely followed the painting project and resulted in none of the painted aircraft being involved in any mid-air collisions. Most pilots commented that the painting technique contributed to the ease at which the aircraft was detected. Fluorescent orange paint was consistently recommended to the FAA to increase sUAS conspicuity.

The Applied Psychology Corporation (1961) conducted a field study on aircraft detection and color identification threshold ranges for the Federal Aviation Agency. Their research observed 541 operational aircraft from the ground to obtain approximate ranges. The research revealed that fluorescent colors could be identified twice as far away as non-fluorescent colors. While research concluded that size is the most important factor, fluorescent paint had a greater range no matter the size of the aircraft. Figure 2 presents threshold ranges of color based on aircraft size.

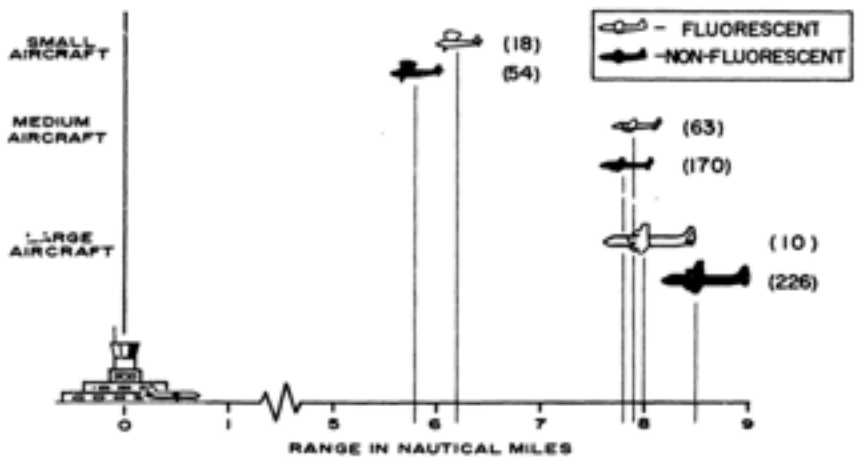


Figure 2. Average threshold ranges for fluorescent painted and non-fluorescent painted aircraft. Adapted from "Field Study of Threshold Ranges for Aircraft Detection and Color Identification" by the Federal Aviation Agency, 1961, Applied Psychology Corporation AD No. 727302.

Research conducted by Hodgson (1959) studied the use of paint to maximize the visibility of aircraft. Hodgson used ground observers, photographs, and three aircraft flying in formation, each painted a different color scheme. The study concluded that two basic principles exist to achieve a high-visibility design. To maximize contrast, black paint should be used in areas that are normally dark or shaded by the plane while areas that are normally bright should be painted white. The second principle to achieve high-visibility outlined in Hodgson's research is the use of fluorescent red or orange paint in areas that are in direct sunlight. These principles are consistent with the Applied Psychology Corporations previous research and commenters suggestions.

Fletcher, Gifford, Lazo, and Siegel, (1969) developed a report on aircraft in-flight camouflage for the Department of the Navy. The report outlines various techniques that can be used to increase aircraft camouflage. One technique outlined in the report is the use of countershading. Countershading is using light paint in the areas that are normally in the shadow and darker paints in the normally highlighted areas. This technique is the opposite of the suggested methods and principles to increase aircraft conspicuity as previously discussed.

Siegel and Lanterman (1963) explored the use of fluorescent paint to increase aircraft detectability and conspicuity. The research centered around studies and pilot opinions on the use of fluorescent paint. Over 91 percent of the pilots interviewed expressed either a mildly positive or strongly positive opinion towards the use of fluorescent paint in increasing aircraft detectability and conspicuity. Table 3 presents the data concerning pilot opinions on the use of fluorescent paint. The research went further to suggest a possible high visibility paint scheme. Siegel and Lanterman (1963) suggested using either a glossy sea blue or white paint in conjunction with a red-orange or orange-red fluorescent paint to increase detectability and conspicuity. The research concluded that fluorescent paint might help increase detectability and conspicuity but more importantly, they concluded that fluorescent paint certainly would not create any harm.

Table 3
Pilot Opinion on the use of Fluorescent Paint

Squadron Type	Strongly Negative		Mildly Negative		Neutral		Mildly Positive		Strongly Positive	
	N	%	N	%	N	%	N	%	N	%
Helicopter	0	0	1	5	0	0	14	67	6	29
Reserve	0	0	0	0	1	3	25	76	7	21
Attack	0	0	1	5	0	0	14	64	7	32
Utility	1	10	2	20	2	20	5	50	0	0
Patrol	0	0	1	10	1	10	5	50	3	30
TOTAL	1	1	5	5	4	4	63	50	23	25

Note. Adapted from "Aircraft Detectability and Visibility: VI. A Qualitative Review and Analysis of the Utility of Fluorescent Paint for Increasing Aircraft Detectability and Conspicuity" by R.S. Lanterman, A.I. Siegel, 1963, AD No. 298331, Philadelphia, PA, U.S. Naval Air Material Center.

The Applied Psychology Corporation (1962) completed an outdoor evaluation of six paint patterns for the Federal Aviation Agency. Pilots viewed different size model aircraft with the six paint patterns on background panels under varying meteorological conditions. The researched concluded that any pattern was better than the aluminum/metallic color model, but none of the patterns were significantly more detectable than each other. The pattern that included white on top, gray on bottom, and a red-orange fluorescent empennage would offer the greatest reliability in a variety of environmental conditions as compared to the other five patterns. Similar patterns have been presented as potential solutions to increase conspicuity.

Research conducted by Bynum, Bailey, Crosley, and Nix (1967) considered paint schemes to improve helicopter conspicuity. The top of the helicopter rotors were painted in six different color schemes to evaluate their ability to increase conspicuity. Table 4 presents the mean first place rankings and *t*-Scores of the selected schemes. Data from Table 5 results in the white and fluorescent red-orange color scheme being statistically significant ($p \leq .05$) when compared to the other color schemes. The six schemes were also studied on two different days with varying amounts of light. Table 6 provides mean first place rankings and *t*-Scores on the two tested days. The white with FRO was consistently ranked first overall, with a mean score of .95 for day one and .98 for day two. Similar to many of the studies previously reviewed, researchers acknowledge that any paint scheme is recommended versus not using a scheme. Another important finding of their research was that they did not identify any significant differences in the rankings between pilots and non-pilots.

Table 4
Rankings of Helicopter Rotor Blade Paint Schemes

Scheme	Mean Proportion	*t*-Score
White – Fluorescent Red-Orange	.9650	68.1190
Black – Gloss White	.6650	54.2610
White – Orange-Yellow	.5400	51.0040
Black – Codit White	.4600	48.9960
Black Tip - White Tip	.3700	46.6810
Non-Painted	.0000	00.0000

Note. Adapted from "Development of a Paint Scheme for Increasing Helicopter Conspicuity" by J.A. Bynum, R.W. Bailey, J.K. Crosley, M.S. Nix 1967, USAARU Report No. 68-1, Fort Rucker, AL U.S. U.S. Army Aeromedical Research Unit.

Table 5

Significance Levels of Ranking of Helicopter Paint Schemes

	Paint Scheme					
	White, Fluorescent Red Orange	Black – Gloss White	White, Orange-Yellow	Black – Codit White	Black Tip – White Tip	Non-Painted
White, Fluorescent Red Orange	-	$p \leq .05$	$p \leq .05$	$p \leq .05$	$p \leq .05$	$p \leq .05$
Black – Gloss White	-	-	-	-	$p \leq .05$	$p \leq .05$
White, Orange- Yellow	-	-	-	-	-	$p \leq .05$
Black – Codit White	-	-	-	-	-	$p \leq .05$
Black Tip – White Tip	-	-	-	-	-	$p \leq .05$
Non-Painted	-	-	-	-	-	-

Note. Adapted from "Development of a Paint Scheme for Increasing Helicopter Conspicuity" by J.A. Bynum, R.W. Bailey, J.K. Crosley, M.S. Nix 1967, USAARU Report No. 68-1, Fort Rucker, AL U.S. U.S. Army Aeromedical Research Unit.

Table 6

Comparison of Rankngs for Helicopter Rotor Blade Paint Schemes on Consecutive Days

		Paint Scheme					
		White, Fluorescent Red Orange	Black – Gloss White	White, Orange Yellow	Black – Codit White	Black Tip – White Tip	Non-Painted
Day 1	Mean Pro-portion	.95	.74	.52	.44	.35	.00
	t-Scores	66.45	56.43	50.50	48.49	46.15	00.00
Day 2	Mean Pro-portion	.98	.59	.56	.48	.39	.00
	t-Scores	70.54	52.28	51.51	49.50	47.21	00.00

Note. Adapted from "Development of a Paint Scheme for Increasing Helicopter Conspicuity" by J.A. Bynum, R.W. Bailey, J.K. Crosley, M.S. Nix 1967, USAARU Report No. 68-1, Fort Rucker, AL U.S. U.S. Army Aeromedical Research Unit.

A FAA report developed by Williams and Gildea (2014) reviewed research related to UAS visual observers. The review revolved around the various human factors associated with the tasks performed by visual observers. The purpose of this review was to evaluate VO requirements to determine if any of these requirements needed to be strengthened or exceed the capabilities of a visual observer. Research highlighted several factors that affect see and avoid capabilities. One particular factor affecting see and avoid is poor contrast. Several suggestions to improve aircraft conspicuity would be achieved by increasing contrast. An increase in contrast can be achieved thru the use of light paint in areas that are normally illuminated and using dark paint in the normally shaded areas. A paint scheme that increases contrast has been suggested by several researchers and by commenters during the development of 14CFR Part 107.

Adamson (1959) reported on United States Coast Guard's efforts to mitigate mid-air collisions. The Coast Guard considered two methods, flight-crew discipline to ensure positive routine out-the-window visual scanning and the use of a high visibility paint scheme. The paint scheme employed by the Coast Guard utilized white paint with fluorescent orange trim. Alternating stripes of white and fluorescent orange paint is an option suggested to the FAA to increase conspicuity while the use of white and fluorescent orange has been suggested by other conspicuity studies.

Hall and Meeker (1995) explored the use of Multispectral Camouflage Appliques (MCAs). MCAs assist in camouflaging by manipulating the spectral characteristics of a high-value asset to disguise these assets as trees or parking lots while simultaneously reducing their thermal contrast. A key advantage associated with the use of MCAs is the ability to rapidly apply any camouflage technique or color scheme that may be required. The use of high visibility MCAs such as international orange, lime-yellow, and bright red have been successfully tested as landing and navigation makers.

Research conducted by Bauhmhardt, Blackwell, Fernandez, and Gaffney (2011) is concerned with the correlation between aircraft color and bird strikes. Their research sought to identify the impact aircraft fuselage might have on the number of bird strikes voluntarily reported. Researchers utilized a red, green, blue (RGB) additive color model as an index to measure the intensity of light in the RGB spectrums. A higher RGB score represents a higher level of intensity or a brighter color while a lower RGB score represents a darker color. The findings suggested that a correlation exists between the front and rear fuselage mean RGB value scores and the bird strike rate per 10,000 movements. Figure 3 illustrates that as the mean RGB value increases, the bird strike rate decreases.

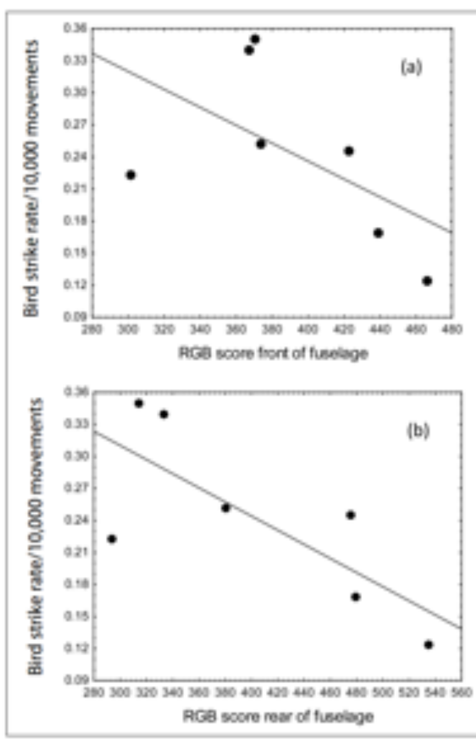


Figure 3. Relationship between bird-strikes and RGB score. Adapted from "Bird Strikes and Aircraft Fuselage Color: A Correlational Study" by P. Baumhardt, B.F. Blackwell, E. Fernandez-Juricic, and E.J. Gaffney, 1961, Human-Wildlife Interactions, 5, p. 224-234.

Their research concluded that bird-strike rates were negatively associated by the RGB score but not in a significant manner (*p = 0.192*). The researchers believe that their methodology could be used to measure avian responses to remote controlled aircraft that utilize different color schemes.

Barrett and Melkert (2005) studied the use of adaptive materials to achieve visual signature suppression to aid in UA camouflage. By tailoring the color and luminosity of a UA researchers were able to reduce the visual cross section of the aircraft to the point it was not detectable to the human eye from the ground. Decreasing the aircraft's illuminance under normal daylight conditions resulted in the darker aircraft standing out more against the white cloud background. The contrast between the dark aircraft and the light background made the UA easily identifiable to ground observers.

Federman and Siegel (1965) preformed five studies to determine what paint schemes would improve aircraft detectability and visibility. Federman and Siegel (1695) defined two thresholds, object and color, for obtaining measurements; (a) object threshold is the maximum distance at which the stimulus could be detected with certainty and (b) color threshold is the maximum distance at which the colors could be identified. The first study evaluated two sets of measurements, the outside limit (object threshold) and the inside limit (color threshold). Data concerning the inside limit measurements indicated that fluorescent blue had the largest visual field. While data on outside limits indicated that fluorescent colors have a larger visual field over ordinary colors. Vision fields of fluorescent and non-fluorescent colors yielded statistically significant differences at the .01 level of confidence. The second study utilized two luminance conditions, high and low, to determine object and color thresholds. Results of the second study indicated that fluorescent paints had a lower color threshold, but fluorescent colors had a higher object threshold as compared to their ordinary color counter parts. An analysis of variance for the data from the second study indicated a .01 level of confidence for colors and luminance levels for both object and color thresholds. Federman and Siegel (1965) conducted two field studies on the detectability of the selected stimuli. Table 7 provides object threshold ranking in field study two for the six different color schemes under varying meteorological conditions.

Table 7
Color Scheme Object Threshold Rank-Order

Rank Order	Sunny A.M.	Sunny P.M.	Cloudy
1	Fluorescent yellow-orange	Fluorescent yellow-orange	Fluorescent yellow-orange
2	Fluorescent red-orange	Fluorescent red-orange	Fluorescent red-orange
3	White	Ordinary orange	White
4	Fluorescent red-orange with a white medial stripe	Fluorescent red-orange with a white medial stripe	Fluorescent red-orange with a white medial stripe
5	White with a black medial stripe	White with a black medial stripe	Ordinary orange
6	Ordinary Orange	White	White with a black medial stripe

Note. Adapted from "Development of a Paint Scheme for Increasing Aircraft Detectability and Visibility" by P. Federman, and A. Siegel, 1965, Journal of Applied Psychology.

The data from the second study suggested that fluorescent yellow-orange and fluorescent red-orange had greater object threshold rankings as compared to the other stimuli. While fluorescent colors displayed relative superiority, there was no significant difference between fluorescent yellow-orange and fluorescent red-orange. Federman and Siegel (1965) recommended the use of unbroken fluorescent red-orange with a secondary area that provides color and brightness contrast. The suggestion of using a paint scheme involving at least two areas painted florescent/aviation orange along with black and white would be best color scheme based on this research.

The U.S Naval Research Laboratory Human Engineering Branch (1955) studied the detectability of colored targets at sea. Researchers painted aluminum spheres with different colors, to include four fluorescent colors. An aircraft flew the established search pattern while observers looked for the painted spheres from an altitude of 700 feet under varying conditions. Observers recorded the distances at which each of the spheres where recorded. Table 8 provides air to water detection distance comparison between the four selected fluorescent colors and the non-fluorescent colors. With the exception of Neon Red, fluorescent colors had a mean average of at least one mile or greater over non-fluorescents. Researchers concluded that fluorescent yellow-orange and red-orange had a greater air to water detection range when compared to non-fluorescent paints, which would be applicable for detecting a seaplane type sUA operating on the water.

Table 8
Air to Water Detection Distance in Miles

Color of Sphere	Number of Observations	Mean Detection Miles
Arc Yellow	24	2.6
Fire Orange	24	2.4
Saturn Yellow	24	2.0
Neon Red	24	1.4
Non-Fluorescent	24	1.0

Note. Adapted from "Field Study of Detectability of Colored Targets at Sea" by Human Engineering Branch, 1955, (Report No. 265). New London, CT: U.S. Naval Medical Research Laboratory

Methodology

A study of archival data is used to identify, gather, and evaluate color(s) or patterns that would help increase small unmanned aircraft conspicuity and assist in achieving and maintaining unaided visual contact by the sUA PIC and also by pilots of manned aircraft. Descriptive statistics including mean rank-order and visual detection range are used to develop a recommendation on the colors or schemes that can be employed by small unmanned aircraft.

Results

From the selected archival data, 11 out of the 16 studies recommended the use of some hue of fluorescent paint. Color schemes that included hues of fluorescent red, orange, or yellow had greater mean detection ranges than other fluorescent colors and enamels of any other color. Fluorescent oranges outperformed every tested color with a mean detection range of 2.35 miles. Color schemes that utilized hues of fluorescent yellow had a mean detection range of 2.3 miles while fluorescent reds had a mean detection range of 1.85 miles. Fluorescent colors had an overall mean detection range of 2.16 miles as compared to non-fluorescent colors mean detection range of 1.0 miles.
Other studies utilized a rank-order system to make determinations on the preferred color scheme to be employed. Fluorescent colors, in general, had a rank-order mean greater than that of non-fluorescents colors. A paint scheme containing elements of red and/or orange were recommended in 10 of the reviewed studies. A fluorescent red and/or orange have a mean rank order of 1.5. Federman and Siegel (1965) was the only study that included a fluorescent yellow-orange hue combination, which resulted in an overall first place rank order. Non-fluorescent yellow-orange or yellow hues were utilized in other studies and would reduce the rank-order to 2.03. Conversely, the addition of all red and/or orange combinations reduces the overall red/orange rank-order to 2.33. The combination of black and white, in various design schemes, resulted in an overall rank-order of 4.03. Of the reviewed archival data, the use of

black and white had the most data points, 14, followed by red and/or orange at nine.

Increasing contrast is another method that is recommended in the reviewed archival data. Six out of the 16 studies recommend maximizing contrast to increase conspicuity by utilizing light colors in normally bright areas and dark colors in normally dark areas. Four of these six studies recommended increasing the effects of contrast by also utilizing elements of fluorescent red-orange.

Analysis

Fluorescent colors outperformed non-fluorescent colors significantly in both rank-order and mean detection range. Fluorescent colors were recommended in 11 of the 16 reviewed studies with 10 of the fluorescents recommended being of red and/or orange hues.

Fluorescent color paint schemes may be a simpler, lighter, and less costly visual detection enabler for mitigating some of technological challenges of see-and-avoid equipage in manned aircraft. Manned aircraft operators are required by 14CFR §§91.225 to equip aircraft with ADS-B (Out) equipment by 2020 (Federal Register, 2015). Technologies including the Automatic Dependent Surveillance-Broadcast System (ADS-B) may be too costly, too heavy, or too difficult to integrate on to all unmanned aircraft. ADS-B uses Global Positioning Satellites (GPS) to report aircraft position and other data (ADS-B [Out]) receive aircraft data (ADS-B[In]) on a cockpit display for traffic information (CDTI) (FAA, 2016). Fluorescent color paint schemes are not an active means of reporting sUA location but could allow pilots of manned aircraft and sUA operators to detect and maintain visual contact with the sUA at a greater distance. An increased visual detection range could improve the ability of pilots of manned aircraft to see-and-avoid, remain well clear, and avoid collision hazards with sUA, as is done through electronic means with an ADS-B(In) CDTI, but at a shorter range.

Maintaining visual contact is one of the key provisions of 14CFR Part 107 and fluorescent colors could enhance a remote PIC's ability to meet these regulations. An example of where fluorescent colors could have an impact on future regulations is the current altitude limitations. Currently, unmanned aircraft have a maximum operating altitude of 400 feet AGL. It may be plausible for Civil Aviation Authorities (CAA) to increase the allowable operating altitude of unmanned aircraft if they are more readily visible. Remote PIC's also have the see-and avoid responsibilities at all times. Fluorescent colors could assist in extending the pilot's visual detection range, thereby increasing the allowable operating range.

The use of sUA in agricultural aerial spraying applications could benefit from the increased visual range associated with fluorescent colors. An operator maintaining a static control position could operate a sUA at a greater visual range and cover more area from one position, potentially removing the need for an operator to relocate to continue or complete spraying operations. In contrast, certain operators may wish to conceal, camouflage, or reduce the visual signature of a sUA. Local police may wish to reduce the conspicuity of sUA for applications requiring stealth and should avoid the use of fluorescent colors. Conversely, first responders may want to utilize fluorescent colors in search and rescue operations.

Fluorescent colors could reduce the potential of bird strikes and reduce any potential environmental impact potentially associated with increased unmanned aircraft operations. As previously highlighted, research conducted by Bauhmhardt et al. concluded that a brighter fuselage resulted in a lower number of bird strikes. Fluorescent colors could help birds identify unmanned aircraft in flight and prevent collisions that may result in the loss of a system and an overall reduction in avian populations.

Conclusions

The use of fluorescent paint schemes on airplane and rotorcraft type small unmanned aircraft, irrespective of their size, increases the visibility, detectability, range, and conspicuity, with fluorescent red-orange as the most desirable color. Fluorescent paint schemes of any color were in all cases an improvement over enamels or an unpainted metallic surface. An increase in sUA conspicuity not only improves the ability of pilots of manned aircraft to detect and avoid sUA's, but also extends the operating range in which the remote PIC and VO can maintain unaided visual line of sight. Although the overall size of a sUA is a factor that contributes towards increasing visual conspicuity, the most desirable color paint schemes were size agnostic.

Recommendations

When considering a paint scheme to increase sUA conspicuity fluorescent colors should be used instead of enamels or unpainted metallic surfaces. In particular, fluorescent red-orange paint schemes covering large sections of the aircraft and/or rotor blades should be utilized. Furthermore, the use of a paint scheme of a black bottom contrasting with a mostly white top and at least two large areas painted florescent red-orange should be considered. Figure 3 is an example of how the recommended paint scheme could be utilized to increase sUA conspicuity.

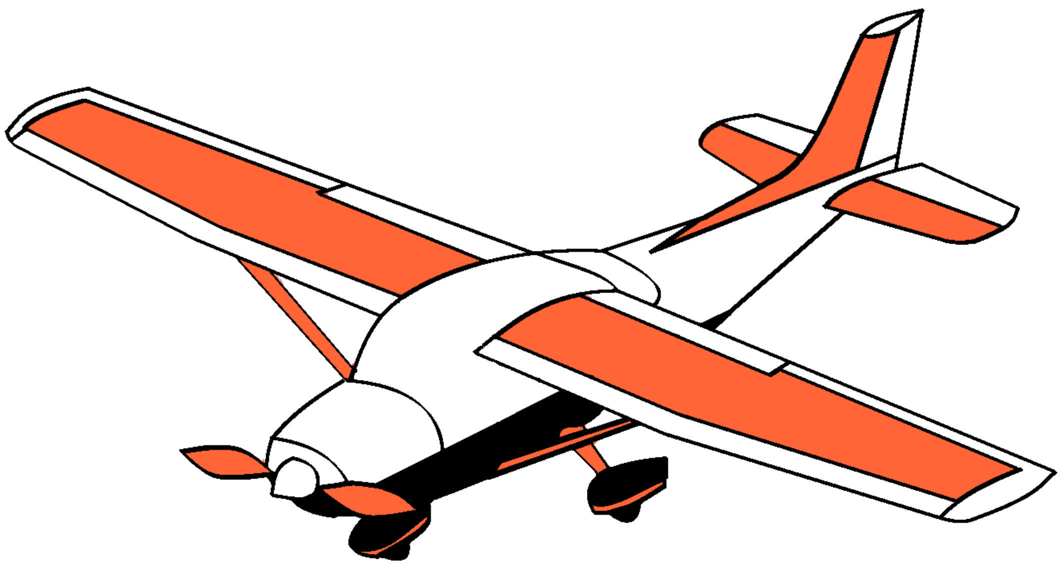


Figure 3. Example of the Recommend Paint Scheme for Small Unmanned Aircraft. A visual depiction of how the recommended paint scheme could be utilized and may vary depending on the UA. Image adapted from Clipart.org (2014).

For sUA mission applications that require occasional stealthy operations, such as law enforcement, fluorescent red-orange Multispectral Camouflage Appliques could be applied to a sUA when an increase in conspicuity is needed and removed when lower conspicuity may be needed. Civil Aviation Authorities should consider the use of fluorescent colors in the development of future regulations and requirements.

REFERENCES

Adamson, R. (1959). *Coast Guard Problems and Techniques of Visual Search in Air-Sea Rescue Operations.* Unclassified.

Bailey, R., Bynum, J., Crosley, J., & Nix, M. (1967). *Development of Paint Scheme for Increasing Helicopter Conspicuity.* (USAARU Report No. 68-1). Retrieved from http://www.usaarl.army.mil/techreports/68-1.pdf

Baker, C. (1960). *Visual aspects in collision avoidance of Air Force aircraft.* Visual Search Techniques., Washington: National Research Council.

Barrett, R. & Melkert, J. (2005). *UAV visual signature suppression via adaptive materials. In Smart Structures and Materials.* (pp. 100-110). International Society for Optics and Photonics.

Baumhardt, P., Blackwelll, B. F., Fernández-Juricic, E., & Gaffney, J., (2011). Bird strikes and aircraft fuselage color: a correlational study. *Human-Wildlife Interactions*, 5, 224-234

Clipart.org (2014). *Airplane*. Retrieved from https://openclipart.org/detail/191525/airplane

FAA Modernization Reform Act of 2012, 49 U.S.C.§ 40101 (2012).

Federal Aviation Administration. (2016). *Automatic Dependent Surveillance-Broadcast (ADS-B)*. Retrieved from http://www.faa.gov/nextgen/programs/adsb/

Federal Aviation Administration. (2016). *Operation and Certification of Small Unmanned Aircraft* (Document No. 2016-15079). Washington, DC: Federal Register.

Federal Aviation Administration. (2010). Aircraft Accident and Incident Notification, Investigation, and Reporting. (Order No. 8020.11C.). Washington, DC: Federal Aviation Administration.

Federal Aviation Administration. (2016). *UAS Sightings Report* [Data file]. Washington, DC: Federal Aviation Administration. Retrieved from http://www.faa.gov/uas/resources/uas_sightings_report/

Federal Aviation Administration. (2016). *FAA Near Mid Air Collisions* [Data file]. Washington, DC: Federal Aviation Administration. Retrieved from http://www.asias.faa.gov/pls/apex/f?p=100:34:0::NO:::

Federal Aviation Administration. (2015). *Automatic Dependent Surveillance—Broadcast Out Performance Requirements To Support Air Traffic Control Service*. (Docket No. FAA-2007-29305) Washington, DC: Federal Aviation Administration.

Federal Aviation Agency, Applied Psychology Corporation. (1961). *The Role of Paint in Mid-Air Collision Prevention.* (AD No. 273691). Retrieved from http://www.dtic.mil/dtic/tr/ fulltext /u2/273691.pdf

Federal Aviation Agency, Applied Psychology Corporation. (1961). *Field Study of Threshold Ranges for Aircraft Detection and Color Identification.* (AD No. 727302) Retrieved from www.dtic.mil/cgi-bin/GetTRDoc?Location=U2&doc=GetTRDoc.pdf&AD=AD0727302

Federal Aviation Agency, Applied Psychology Corporation. (1962). *Outdoor Test Range Evaluation of Aircraft Paint Patterns.* (AD No. 407). Retrieved from http://www.dtic.mil/ dtic/tr/fulltext/u2/407910.pdf

Federman, P. & Siegel, A. (1965) Development of a paint scheme for increasing aircraft detectability and visibility. *Journal of Applied Psychology, 49*(2), 93.

Fletcher, D., Gifford, E., Lazo, J., & Siegel, A. (1969). *Guide to Aircraft In-Flight Camouflage.* (No. NADC-AC-6904). Wayne, PA: Applied Psychological Services Inc Science Center.

Gildea, K. & Willams, K. (2014). *A Review of Research Related to Unmanned Aircraft System Visual Observers*. (No. DOT/FAA/AM-14/9). Washington, DC: Federal Aviation Administration Office of Aerospace Medicine.

Hall, K. G., & Meeker, D. L. (1995). *Fast application multispectral camouflage appliques*. Proc. SPIE 2497, Public Safety/Law Enforcement Technology, 105; doi:10.1117/12.210487

Hodgson, W. (1959). *High-Visibility Painting Design for Aircraft* (No. NEL-910) Navy Electronics Lab San Diego CA.

Human Engineering Branch (1955) *Field Study of Detectability of Colored Targets at Sea*. (Report No. 265). New London, CT: U.S. Naval Medical Research Laboratory.

Lanterman, R., & Siegel, A. (1963) *Aircraft Detectability and Visibility: 6. A Qualitative Review and Analysis of the Utility of Florescent Paint for Increasing Aircraft Detectability and Conspicuity*. (AD No. 298331). Philadelphia. PA: U.S. Naval Air Material Center.

King, J. G., & Solomon, S. S. (1995). Influence of Color on Fire Vehicle Accidents. *Journal of Safety Research*, 26(1), 41-48. doi:10.1016/0022-4375(95)00001-1

Rasmussen, P., Vaughan, J., & Welsh, K., (1978). *Conspicuity assessment of selected propeller and tail rotor paint schemes*. United States Retrieved from https://babel.hathitrust.org/

U.S. Department of Transportation, Federal Aviation Administration (2016) *Small Unmanned Aircraft Systems (sUAS)* (AC No: 107-2)

JOURNAL OF UNMANNED AERIAL SYSTEMS

PEER-REVIEWED ARTICLE

CYBER ATTACKS AND DEFENSE FRAMEWORK FOR UNMANNED AERIAL SYSTEMS (UAS) ENVIRONMENT

Kevin Casagrande
Joshua Friederichs
Clarissa Gonzalez
Tanya Humphries
Zachary Tindall
Undergraduate Students, Electrical Engineering, University of North Dakota

Roger French
Principal Engineer, Rockwell Collins Corporation, Cedar Rapids, Iowa

Dr. Prakash Ranganathan
Assistant Professor, Electrical Engineering, University of North Dakota

ABSTRACT

The proliferation of privately-owned Unmanned Aerial Vehicles (UAVs) and the associated potential risks they pose to personal privacy and safe navigation has highlighted the need for an effective system to test UAV cyber security vulnerabilities and develop effective countermeasures. Towards those ends, we present a novel, customizable, and a universal UAV simulation environment utilizing an open-source Linux-based virtual machines (VM) for modeling cyber-attacks and defenses against ArudoPilot-MavLink-based fixed-wing and rotary-wing UAVs. To demonstrate the effectiveness of this simulation environment, the percentage of successful man-in-the-middle and re-broadcasting attacks conducted by a Kali Linux VM's against a stock ArduPilot UAV were examined. Additionally, the simulation environment was used to develop and test a unique defensive system based on an external SNORT intrusion detection system (IDS) to monitor communications between the simulated UAV and ground control system (GCS). Upon intrusion detection, this system automatically executed pre-planned countermeasures to defeat potential threats. The effectiveness of this system was evaluated based on the reduction of successful cyber-attacks and vulnerabilities and developing effective countermeasures without the time and cost required for physical prototyping using actual UAV assets was effectively demonstrated.

I. Introduction

With the rise in commercial and military use of unmanned aerial vehicles (UAVs or "Drones"), a lot of research has been conducted on the potential vulnerabilities inherent in autonomous technologies which operate geographically detached from human operators (Paganini, 2013). Most notably, Todd Humphreys and his team at the University of Texas have made considerable strides in both GPS spoofing and detection or erroneous GPS signals and the application of these technologies to UAVs (K, 2015).

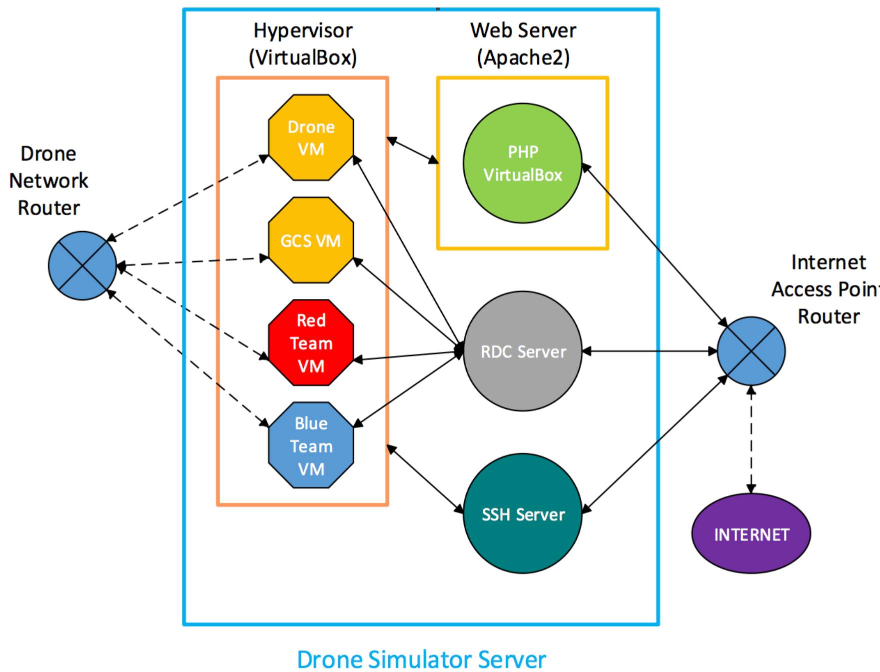


Fig. 1 – System Diagram of Drone Cyber Attack and Defend Simulator

Javaid et al in (Javaid, Sun, Devabhaktuni, & Alam, 2012) published a comprehensive analysis of vulnerabilities in drone design and the potential impacts on vehicle operation. However, with the commercial applications of drone technology beginning to dwarf the military applications, the likely targets of cyber-attacks are not large, military grade, hardened UAVs but rather the small, cheaply produced, and easily operated drones favored by commercial enterprise. Cyber security specialists and hacker collectives alike have devoted extensive analysis resources towards this area, as evident by frequent presentations at Def Con and similar conferences. In addition, some academic teams have started performing research on this topic. Hal Aldridge and a team from Purdue University developed a simulator and modelled the behavior of a UAV exposed to several forms of cyber-attacks (Kim, Wampler, Goppert, Hwang, & Aldridge, 2012). They proposed developing a method to detect and counter those attacks.

II. System Description

The project focuses on the development of open-source systems conforming to the specifications of the Dronecode project (www.dronecode.org) and ArduPilot (http://ardupilot.com/) to develop a VM-based drone simulation suitable for testing a wide array of cyber security tools and techniques. A basic block diagram of the system is provided in Fig 1. The system consists of a virtualization server which hosts the 4 Virtual Machines (VMs). These are the Drone VM, the Ground Control Station (GCS) VM, an attack VM and a defense VM. The main element of the system is a virtualization server composed of an Ubuntu 12.04 headless server running Oracle VirtualBox. A web interface for control and operation of the virtualization server is accomplished using PHPVirtualBox hosted through an Apache2 web server installed on the virtualization server base operating system.

The first element is the Drone VM, which is an Ubuntu 14.04 based host for the ArduPilot drone operating system. The ArduPilot code is fully compliant with the DroneCode standard and is the actual software which would run on physical drones conforming to this standard. The Drone VM includes both the fixed-wing and helicopter variants of this software. Additionally, the Drone VM is equipped with the JSBSim flight simulator software, which is used to provide the flight dynamics for the drone simulation. Finally, the Drone VM includes the MAVProxy software, which it uses to provide a UDP-based communication circuit for the drone. An overview of the drone VM can be seen in Fig. 2. Using these three elements in tandem allows for the accurate simulation of a wide-variety of fixed-wing and rotary type drones in user-programmable environmental conditions. Although the ArduPilot software would normally be constrained by the physical design of the drone it was running on, these parameters are user-programmable in the simulated environment. Thus any collection of drone parameters the user wishes to use can be simulated. This set-up also allows for follow-on efforts for hardware-in-the-loop testing utilizing a physical drone controller and actual live-testing of a complete drone system.

Simulation of the drone controller or operator is accomplished through the GCS VM. Like the Drone VM, the GCS VM is based on Ubuntu 14.04 and hosts the MAVProxy software. In the case of the GCS, however, the MAVProxy software is used as a ground control system which communicates with the ArduPilot software on the Drone VM via either Ethernet or Wi-Fi communications channels. The user inputs commands to the drone on the GCS VM, which are then communicated to the Drone over the network connection and the ArduPilot software executes these commands. The feedback to the ArduPilot is provided by the JSBSim software, which simulates the drone behavior. Through this communications link, the drone position, speed, and other characteristics are fed back to the GCS VM. This is in effect the same operation as if the operator were communicating remotely with an actual drone. The design of the system allows for any compatible ground control software and/or physical controller to be used to control the simulated drone.

To demonstrate the validity of this system, and for identification and correction of potential cyber vulnerabilities, this project includes additional VMs to demonstrate the use of offensive and defensive techniques to test and improve the simulated drone system. Currently two defensive VMs are included on the virtualization server: one based on Ubuntu 14.04 desktop and a second utilizing the SecurityOnion operating system. These VMs operate almost identically to a physical machine operating on the same network as the drone and ground control station. For testing offensive cyber tools, a fifth VM is currently installed which is a Kali Linux 2.0 operating system. Since these VMs operate on the same drone network as the Drone VM and GCS VM, and since the packets are physically routed from these VMs to a common router via separate adapters, the defensive and offensive VMs are able to interact with the drone and GCS identically as if they were physical machines. This means intrusion detection scans will trigger just as they would in an actual operating environment. Likewise, packet analysis and spoofing attacks will behave just as they would had they been conducted on an operational network.

III. Red And Blue Team Responsibilities

To accomplish the goals of this project, members were divided between three main task groups. Kevin Casagrande is responsible for the design, creation, and maintenance of the simulator environment. Kevin has created the virtualization server and web host as described above. Other responsibilities focus on improving this system and incorporating the hardware elements and physical capabilities required to accurately simulate the offensive and defensive cyber security capabilities utilized by the other groups.

The defensive team (Blue Team) is composed of Tanya Humphries and Joshua Friederichs, is focused on defensive cyber security operations. This group is responsible for testing and implementing tools to prevent the control or destruction of the drone by malicious actors. Thus far, Tanya has focused on Snort and determining how this intrusion detection package can be leverage to identify unauthorized access to the drone and/or GCS and trigger defensive countermeasures. Snort is a sniffer program which captures network traffic and then analyzes the data (S, 2003).

Josh has focused on the Security Onion suite to determine which attacks it can protect against and if it could be suitably tailored to project. Together, Tanya and Josh will decide upon the best configuration for the defensive VM and integrate them into the simulated environment. A summary of the defensive tools is shown in Fig 3.

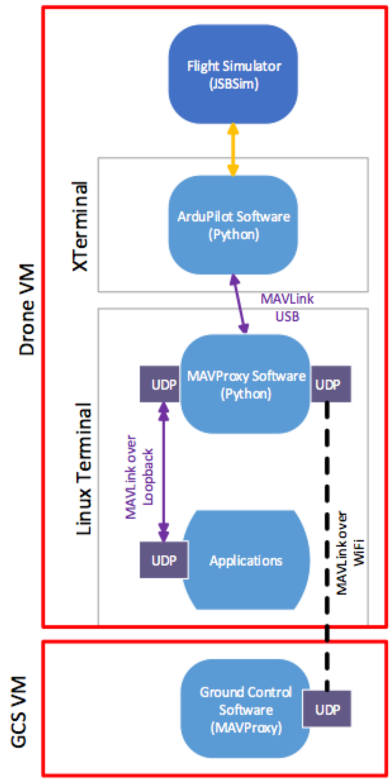


Fig. 2 – Drone Simulator VM Block Diagram

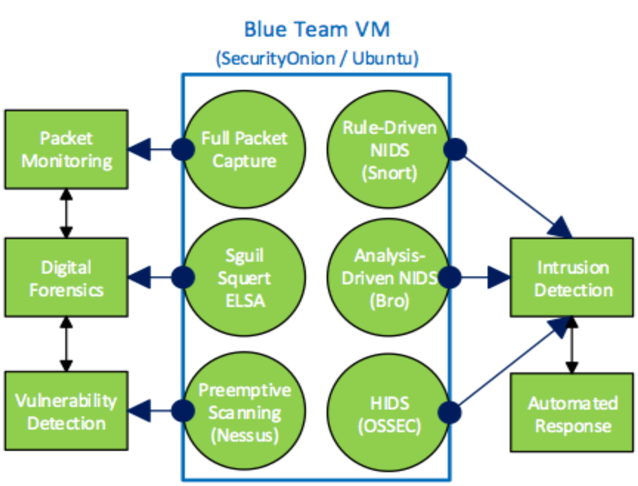


Fig. 3 – Blue Team Block Diagram

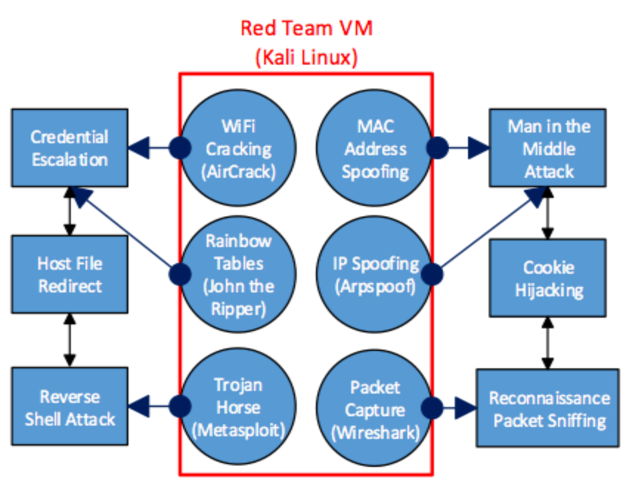


Fig.4. Red Team VM Block diagram

The Blue Team has been able to create a VM to serve as a defensive box that can detect network attacks and defeat them in an automated mode. Open source tools were used to accomplish this to allow for configuration and editing of these tools, as well as economic considerations. The tools making up the defensive VM include Snort, Kismet, Swatchdog, and SendIP.

The attack team (Red Team) is composed of Clarissa Gonzalez and Zachary Tindall. They are focused on offensive cyber security operations and will conduct attacks against the drone and/or GCS with the goal of seizing control or destroying the drone. Simply preventing the drone from executing its tasks will not be considered a successful attack for this project. Both Zach and Clarissa have thus far worked with Kali Linux, as this is likely the most suitable attack platform. Zach has been working on Wi-Fi attacks such as those presented in (Buchanan & Ramachandran, 2015) and (Lakhani & Muniz, 2013), including those aimed at severing the established link between the drone and the GCS and defeating WPA encryption. Clarissa has been working on IP and MAC address spoofing. Through the combination of these efforts, they plan to devise a mechanism for inserting the attack computer in the middle of the communications link between the drone and the GCS without alerting the operator or the Blue Team. Ultimately, Clarissa and Zach will determine the configuration and attack profile most likely to yield success and integrate them into the attack VM for testing. An overview of the tools being used by the Red Team can be seen in Fig. 4. So far the Red Team is executing basic Wi-Fi attacks using airmon-ng, a deauth message, wireshark, and aircrack-ng. The process for this type of attack is shown in Fig. 5.

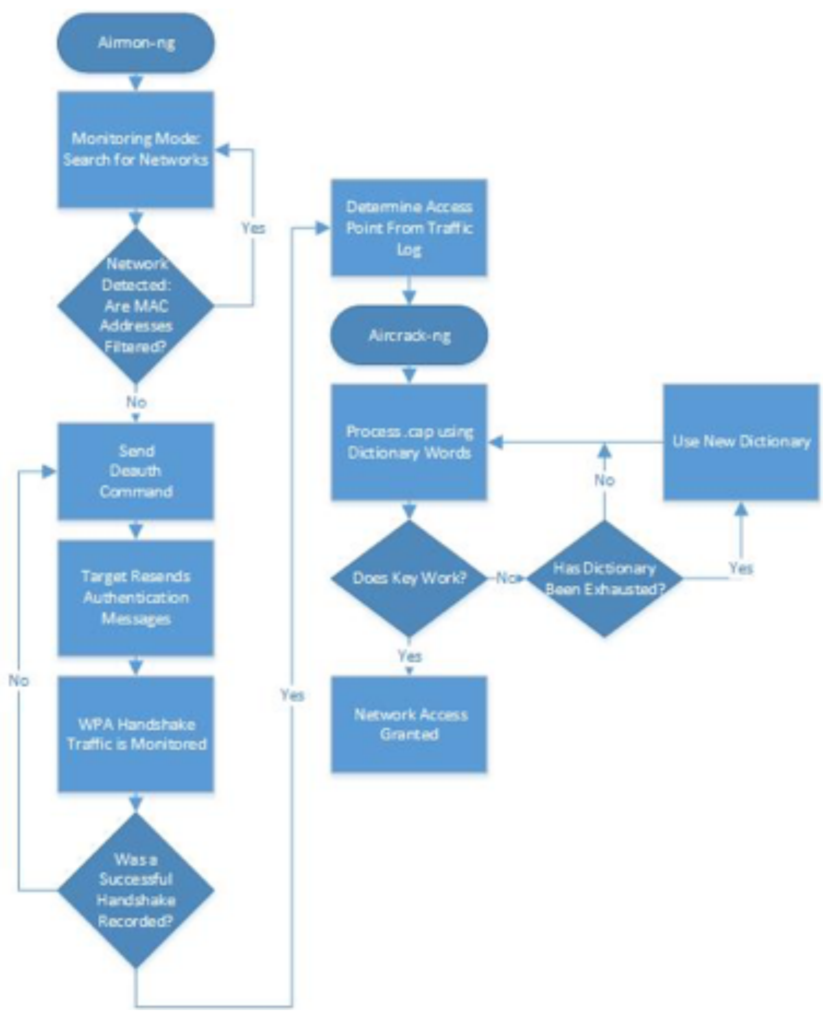


Fig. 5. Flowchart for Wi-Fi Attack

Once both the Red and Blue teams are ready, the project team will test the effectiveness of the defensive configuration against the offensive attacks. From the results of this testing, the group will attempt to improve the robustness of the drone's defenses and identify the likely threat vectors.

IV. Project Tasks

The following tasks have been identified and accomplished for this project:

1. Created a cloud-based VM
2. Developed a web-based interface for an Ardupilot to simulate drone behavior.
3. Ported the drone simulation to a physical device.
4. Developed an ad-hock 2.4 GHz Wi-Fi network interface between the controller VM, Penetration VM, and drone simulation.
5. Tested the effectiveness of attacks on the Wi-Fi connection developed by the Red Team and defenses of the Wi-Fi connection developed by the Blue Team.
6. Conducted the literature review of techniques for attacking Wi-Fi connections, Linux operating systems, controller computers, and drone control software
7. Selected offensive tools and develop familiarity and skill in their use.
8. Developed the attack box (VM or computer) and install required tools.
9. Tested the success of identified attacks, scans, etc. against the cloud-based simulation.
10. Identified Wi-Fi attacks to be tested against the physical Wi-Fi connection.
11. Researched techniques for intrusion detection and prevention for Wi-Fi connections, Linux operating systems, controller computers, and drone control software.
12. Selected defensive tools and develop familiarity and skill in their use.
13. Developed the defense box (VM or computer) and install required tools.
14. Evaluated the effectiveness of defensive tools against the red team attacks.

V. SIMULATION ENVIRONMENT AND SET-UP

The initial part of the project was to design an actual simulation environment. The ultimate end product had to be accessible remotely and capable of hosting simultaneous access to multiple participants. Initially the use of commercial cloud services, such as Digital Ocean and Amazon Web Services, were considered as the primary deployment vehicle, mostly due to the commercial grade connectivity these services provide. However, as research into the requirements for the simulation itself progressed, it rapidly became apparent that the dedicated physical memory and processing required would make any commercial cloud-based service cost prohibitive for use in this project. Ultimately it was decided dedicated server hardware would be required. Privately-owned hardware was repurposed to provide the necessary physical back-bone for the virtualization environment. The end result of this effort was the server construct described above. Physical construction required 36 hours of assembly and configuration.

As commercial web-hosting was not feasible, a dedicated web-interface needed to be developed for hosting remove access to the hypervisor to allow group members to remotely create, start, and stop VMs as necessary for work on the project. As cost was a factor, open source options were favored. Towards that end, a headless Ubuntu 12.04 server was selected as the hypervisor operating system. On top of that OS, Oracle VirtualBox was compiled from source code for use as the hypervisor. Additionally, Apache2 Web Server was selected as the web interface host. Squid proxy server was installed to allow for cacheing of internet data which has the effect of speeding up updates to the multiple VMs by archiving frequently accessed material. To facilitate remove access of VirtualBox services, without having to resort to the command line, PHPVirtualBox was installed and configured as a script-based interface operating on the Apache2 web server. The set-up, installation, testing, and debugging of this set-up required and additional forty-eight hours of dedicated development time.

Once the hosting environment was completed, development of the actual VMs for use in this environment began. To streamline operation of the VMs, each was initially prototyped on a separate physical computer. Once the configuration of each was finalized, rather than transport each VM over to the hosted environment, the VMs were re-created within the virtualization server environment. This ensured a more polished and streamlined deployment of each VM within the server environment, as the baggage and overhead from prototype development and tweaking was not transported over into the new environment.

Development of the Drone VM was particularly challenging, as the overall intent of creating a realistic system analogous to a physical drone was of primary importance. Various attempts at hosting a virtualized ARM-based machine within the VirtualBox environment were attempted with little success.Ultimately the conclusion was reached that in order to accommodate the processing power and virtual memory required to run the flight simulator software effectively, a full Ubuntu implementation would be required. In effect, this artificially increased the processing power and virtual memory allocated to the drone well beyond that typical for a physical model, as both the simulator software and drone operating system had to be installed on the same VM to allow flight dynamic data to be processed without excessive delay. The second major challenge in the development of the Drone VM was in the integration of the ArduPilot software and the JSBSim flight simulation program. Organically, JSBSim would function adequately with the ArduPlane software, but flight dynamics from JSBSim would not act compatibly with ArduCopter operating systems. As any extension of the simulation project to a physical drone would likely involve a multi-copter drone vice a fixed-wing UAV, and as the parameters of the project called for accurately simulating a significant range of potential drone designs, this was deemed to be too limiting of a condition to be considered acceptable. Custom scripts were programmed in order to allow for flight dynamic data from JSBSim to interact with ArduCopter software. While functional, this set-up is still significantly limiting. Therefore, alternative flight simulation systems were examined throughout the duration of this project in parallel with continued development of the primary system using JSBSim. None of these systems improved performance sufficiently to justify the cost of a commercial product.

Additional VMs for dedicated use as ground control, defensive cyber security testing, and offensive cyber security testing were also created as described in other sections of this report. While MAVProxy over MAVLink is the primary ground control software design being used in this project, a number of alternative configurations were also examined. It was determined, at a minimum, that QGroundControl and APM Planner interfaced effectively with the Drone VM, even when operating on Linux and Windows 7 VMs respectively. Some minor testing was conducted using iOS and Android based ground control applications with favorable results. As these are more suited for Hardware-in-the-Loop (HITL) applications, rather than the limited value they present in a virtualized Android development environment, little work was done beyond demonstration of theory-to-concept.

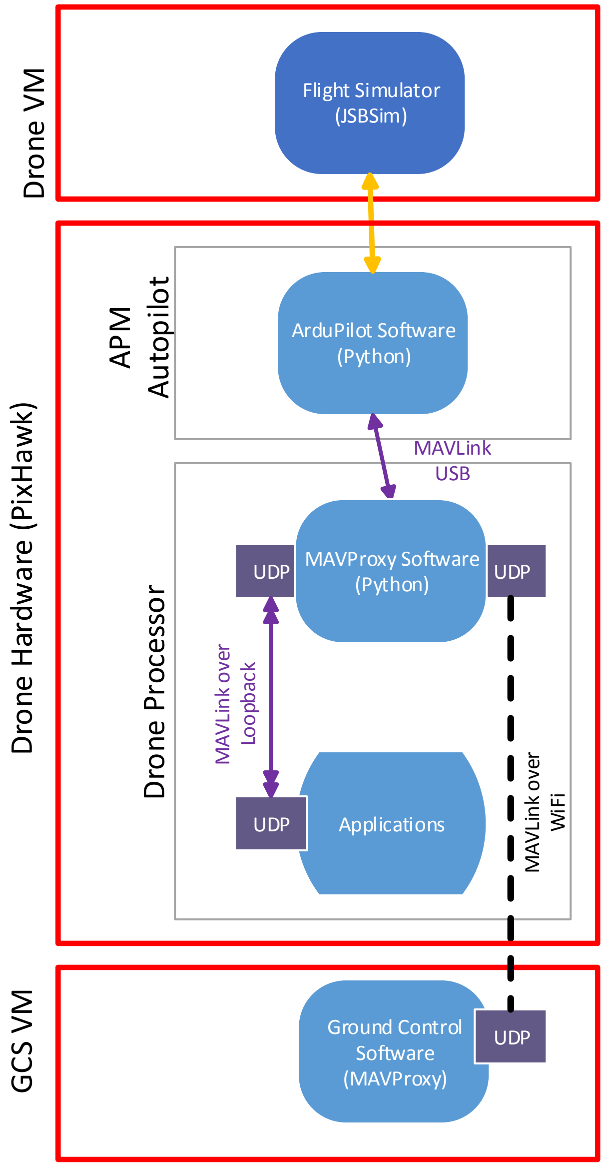


Fig. 6 – Hardware-in-the-Loop (HITL) Block Diagram

For sake of simplicity and standardization, the project has assumed MAVProxy to be the ground control system for testing purposes. This eliminated the need for group members to familiarize themselves with multiple ground control systems.

a) Remote Hosting.

As remote hosting of the environment had to be performed over a residential—vice commercial—internet service, there were a number of challenges in preventing the Internet Service Providers filtering of web traffic. As the type of dedicated traffic to a residential connection tends to trigger ISP anti-spam and anti-Denial-of-Service algorithms, a mechanism for authenticating traffic from group members was required. This was initially accomplished through a very basic PPTP virtual private network setup. However, as this was a crude solution, an improved connection was established using an alias on the Fully Qualified Domain Name (FQDM) associated with this project to forward standard HTTP traffic to a non-standard port number. Initially the interface software for remote access included in

PHPVirtualBox was used to access and control VMs directly. However, this method resulted in an unacceptably high failure of connectivity and furthermore required hard stops of the VMs to resolve the loss of synchronization each time, posing a high risk for corruption of the VMs themselves. Dedicated ports for Microsoft Remote Desktop Connect protocol access were established, using the RDP server services inherent in VirtualBox vice those in PHPVirtualBox. This provided for a very stable connection to the VMs, but still suffered from the lag due to the limitations of the residential internet connection.

An OpenVPN server operating on the virtualization server external router was attempted as a solution, but performance problems plagued this configuration during testing. A dedicated OpenVPN server VM was developed to overcome these limitations. While this configuration was successful for a period of time, ultimately the OpenVPN server VM failed to maintain a connection from outside the domain upon which it was hosted. Significant troubleshooting was successful at resolving these problems. As an attempted solution, the OpenVPN server was rebuilt from scratch several times, with careful attention on kernel configuration during each attempt. In each of these cases, the OpenVPN server functioned to provide a link to the internal drone network from external internet domains, but more advanced applications (such as remote VMs interacting on the network) were extremely limited. This was especially true when more than two remote machines attempted to connect to the network over the VPN connection simultaneously; under these conditions, the connection held but was all but useless from a practical standpoint. Ultimately it was concluded the combined limitations of the open source version of the OpenVPN software and the service provider limitiations placed on a residential internet connection were too great to be overcome by tweaking the OpenVPN configurations.

Given the time delays imposed by remote connections to the virtualization server and hosted VMs, any future iterations of this project will need to consider commercial grade internet hosting (via a university or other entity capable of providing such services), restricting access to a more local geographic area in the proximity of the server, or shifting to an entirely contained local network (i.e. eliminating remote access via the internet) in order to overcome these limitations.

As a potential work-around for the lack of an effective VPN system, archives of VMs were exported from the server into standard OVA files and distributed for individual use by group members. This way, it was believed, group members could work on their local hardware using these VMs, work on their individual segment, and then upload their work for incorporation on the virtualization server for incorporation into the larger project environment. It was hoped this would yield productivity advantages, as the internet lag from remote operation of the VMs would be limited to testing in the full environment. Towards that ends, automated scripts were developed which would archive VMs after significant changes were identified in those VMs. Archiving was accomplished by exporting the VMs to all-inclusive OVA files. A second set of automated scrips automatically synchronized with a dedicated DropBox folder, uploading revisions to the OVA files automatically. As group members had common access to this DropBox folder, they could then download the OVA files as needed for use on their local machines. This also provided a mechanism for establishing regular back-ups for the VMs, should any part of their testing or use cause irrevocable harm or corruption to them.

Extensive component and system testing was required to bring the system as described above to its ultimate state. The initial challenge was establishing the virtual drone construct, as the ArduPilot software is not designed to run on a standard operating system, let alone inside a virtual machine. Once a viable solution was found for these challenges, the resulting drone simulation had to respond appropriately to local commands. Testing therefore consisted of simulating both a quad copter and a fixed wing drone and entering appropriate commands directly into the local MAVLink communications protocol running on the simulated drone. Each of the appropriate commands was entered for each of the drone types a total of ten times in random order. The MAVLink PWM communication to the simulated motors was monitored to ensure expected response to each command. Additionally, the modeled drone behavior in JSBSim in response to each command was monitored for proper simulation as each command was executed. Testing was not considered successful until 100% of drone commands resulted in the expected response in both MAVLink PWM communication and JSBSim simulation results.

b. Communication between Drone and Ground Control Station (GCS).

The second phase of simulator testing involved communication between the simulated drone and a remote ground control station. Since ultimately the intent of the project was to allow for the GCS VM to act as the remote operating station, the test was established to monitor the successful communication between the simulated drone and the GCS VM. Both a quad copter and a fixed wing drone were simulated on the Drone VM. The MAVProxy GCS software on the GCS VM was established as the controlling station associated with each drone. Commands were passed from this controlling station to the drones, such that each command applicable to the drone type was initiated ten times over the course of a two-hour run. During this time, the drone was monitored for proper response to each command. It was observed that 100% of the initiated commands were executed by the drone. Additionally, each command was attempted when the drone was in a powered-down condition to verity negative response. In all such cases, the drones responded as expected. Additionally, throughout the 2-hour run time, packets transmitted/received by the drones and the MAVProxy software were monitored to determine the extent of packet loss. 99.9% of transmitted packets were received, which is appropriate for the UDP protocol used by MAVLink. In fact, this result may be someone unrealistically high, as environmental interference may be significantly higher in an actual drone, depending upon its location and proximity to the controlling station.

During the communication test described above, the synchronization between the drone and the MAVProxy plot of the drone to determine GPS error. At specified intervals, the drone's reported GPS position and the MAVProxy's determined position for the drone were recorded and compared. 73% of the time the two GPS locations were identical. And even when they differed, the difference was less than ½ of the level of accuracy possible given drone speed and update interval. Thus the drone position and MAVProxy plotted position never differed by more than the distance the drone could travel in a single update period, making the error solely a function of the frequency of communication between the drone and MAVProxy. Thus these results approximate the best case for actual drone performance, as the design of the system suggests some additional error between the predicted drone position and it's actual position is likely, especially when environmental factors such as wind are shifting rapidly or of high magnitudes (i.e. wind speed approaching 70% of drone speed at a 30 degree angle off drone course).

c. Interface Testing and Interoperability.

Three major interfaces between the server environment and outside operators (via the internet) were tested for security and operability. First, the web hosting of the PHPVirtualBox interface was tested by attempting to access each of the established accounts from different device types (Windows computer, Ubuntu computer, iPhone, iPad, and Android tablet) from different external domains (Starbucks hosted WiFi, Verizon hot spot, AT&T data services, and GoWifi). For each device, on each internet provider, each account was used to logon to the PHPVirtualBox interface and then used to start, freeze, resume, and stop each VM. In all cases, no abnormalities or errors were observed during these tests. Similar testing was performed on the Secure Shell Server, using a variety of applications from each of these external internet hosting services. For example, PuTTY was used for the windows machine, Juice SSH on the android tablet, and SSH Term on iOS devices. The internal SSH command line interface was used for the Ubuntu machine. Again, in all cases, each account was able to remotely logon to the virtualization server via SSH. Finally, the ability to remotely connect to each VM via Remote Desktop Connect was tested using the RDC program on a windows machine, Remoter Pro on iOS devices, and RDC Connect on the android tablet. As before, these tests were made from the various internet hosting services listed above. While the delay was more pronounced for public hotspots, due to a combination of high loading from multiple users and lower quality WiFi standards due to maximizing operability with all kinds of devices and users, connections were manageable in all cases. However, it was noted from other group members that increased geographic separation greatly amplified the observed delay. The testing was repeated from a 3000-mile geographic separation and the resulting delay was well below the 60 Hz refresh rate considered acceptable (observed refresh was as low as 0.1 Hz on a public network at 3000-mile geographic separation). However, mean refresh rate was 10 Hz when more ideal hosting environments were selected, which is manageable but far from ideal. As previously discussed, increased performance will require either reducing the geographic separation or increasing the bandwidth for the server well beyond that typical of residential internet services.

Security of the server was tested using a Nessus scanner and OpenVAS 8.0 automated vulnerability scanning tools. Some vulnerabilities were initially discovered, but all were successfully patched. Subsequent scans with each subsequent update were performed; no critical vulnerabilities were identified during any of these subsequent security scans.

d. Red Team Goals.

The goal of the Red Team was to demonstrate an attack using the various tools researched for this task. Research into the tool suite in Kali was key to understanding different approaches in analysis of a target, intrusion of the target, and taking over the target. After familiarization of the tools was achieved, Zach was able to start working within the virtual environment that was set up. This environment included the Kali Virtual Machine (VM), the Ground Control Station VM (GCS) and the Drone VM (UAV). Knowing and understanding this environment was key to the following steps that were designated as goals to take over the drone:

- Find IP addresses of Ground Control Station (GCS) and Drone (UAV)
- Find likely ports of communication that command packets were using
- Perform Man in the Middle attack using information above, taking control of UAV.
- Command UAV

The methods of gaining an IP address and router information were initially very useful as it provided a starting point to work with. To cut down on initial work, certain control variables were set. These variables were the IP addresses of the UAV and GCS, the port numbers they were working on, and the protocol they were using. Using these variables, team was able to develop a script that takes command of the UAV. The tools used for this are, in the order of usage, as follows:

- IP forwarding "on" (Kali inherent functionality)
- Scapy ARP spoofing (scapy (P, 2010) is a python-based packet manipulation program)
- IP forwarding "off"
- ifconfig Kali settings set to GCS settings
- mavproxy started and control is taken

The goals of finding the IP addresses of the GCS and UAV and their ports that the command packets are sent were achieved through a combination of the tools researched. To find the UAV and GCS IP's was a matter of scanning for UDP packets given a range. The special part about the UDP packet scan is that it reveals otherwise unknown IP addresses exclusively using the network for UDP messages. The range of IP addresses was the factor that determined how long the scan would take and could be anywhere from a few seconds to hours long. In a real-world situation, reconnaissance would be done before the attack would take place so this would be a low to medium issue.

The way the scan works is that it sends UDP command packets to every IP address listed in the range specified. This scan also attempts to send the UDP packets on a closed port, selected intuitively. This scan then waits for return packets. Upon packets returning with a fail, because the port would be closed and the packet rejected, the IP is revealed to the scan and shown to the user. The same method applies for port scanning as well but this time it is scanning for ports that are in use by the GCS and UAV. This port scan could take a very long time as the ports being used by the GCS and UAV could be changed. However, an unsuspecting client or someone in the unknown would not know that the default port number is specified on their website thus narrowing down the range significantly. This is done manually thus far to incorporate this scanning process into the script designed below.

e. Monitoring Packets through Scapy.

Scapy is a new tool that red team was working with and was chosen on account of more control over what packets are sent, the number of packets sent, python compatibility, and less commands and memory used by the python script. Knowing the UAV and GCS's IP address and the GCS command port, team was able to develop a python script that uses scapy IP spoofing, ifconfig settings, and the mavproxy commands to take control of the UAV. Mavproxy, the protocol used for two years by the PIXHAWK MAV project ("MAVLink Micro Air Vehicle Communication Protocol – QGroundControl GCS," 2015), is used to send commands to the UAV. The UAV initially starts with its output directly forwarded to the GCS's IP and port. This establishes a 'master' via wireless or wired using mavproxy. When this happens the user at the GCS can send commands to the UAV and the UAV follows them as if someone entered the command directly from the UAV's microcontroller.

The script takes the GCS's IP, the UAV's IP, and the interface used for the communication, limited to 'eth0' for now, and starts forwarding the IP address immediately. This gives the Kali user the ability to be the man in the middle. From here, the script automatically goes through the steps to insert itself as the 'master' user for the UAV. This includes enabling and disabling IP forwarding, severing the link between the UAV and GCS. While the GCS and UAV are trying to reestablish a link, the script sets the Kali computer as the 'master' using a subprocess call to mavproxy. Once this is done the user will be able to see heartbeat messages from the UAV and is able to send commands to the UAV. This whole process is automated through Python scripting and tools within python.

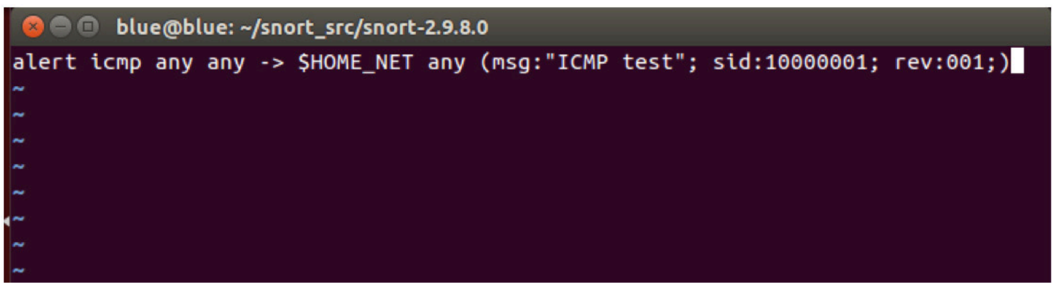


Fig 7. Snort rule file

The tools and code used to script in Python are as follows:

- Importing scapy.all for scapy features
- **echo** command that inserts and saves to a file
- **send** command that sends a packet (scapy tool)
- **subprocess** call that starts a subprocess

These are the main commands that were used along with different parameters. From these tools and commands, three functions were built that accomplish the tasks outlined above. The first is the spoofing function that uses **send** to send ARP packets to the GCS and UAV to actively spoof them. The second is the restoring function that, once the program is closed, restores targets and the system back to default using the same command. The third and final function is the main function where the various other functions are called and the actual attack occurs.

The first task the script does is ask for the IP addresses and the Interface. Once the script has these, the process of attacking begins by port forwarding using the **echo** command, spoofing using the first function, severing the link by disabling port forwarding, and taking control with the **subprocess** call to **mavproxy**. This achieves the goal of demonstrating a man in the middle attack.

Further future development is required in the form of target analysis and wireless simulation. Because the simulation is using a wired, Ethernet connection the method of gaining control of the UAV is different than what it would be in real-world applications. There is also much to be said in terms of "fast" target analysis. The processes of uncovering the GCS and UAV's IP addresses and ports being used have been developed. The only problem with these processes is the length of time it could potentially take to scan for all possible targets. There is also further work to be done in automation of the analysis, but that task is a stretch goal. Overall, the tasks and goals set by the Red Team have been met.

f. Blue Team Goals.

The goals for the Blue Team involved researching the potential defensive tools. Team explored the possibilities of using Snort on Ubuntu tools for defense. Snort is a sniffer application that will monitor all network packets and will compare them to user generated rules. The initial set up of Snort is intensive and required several attempts to get the configuration files edited correctly and get the program running.After successful installation of Snort it was necessary to learn how to write rules and test them. The basic format for a Snort rule is:

action protocol src_ip src_port direction dst_ip dst_port (rule options)

A sample rule can be seen in Fig 7. This rule was used, with some modifications at different times in the testing to verify the system responses. The actions available in Snort are alert, log, pass, activate, dynamic, drop, reject, and sdrop. So far the actions the team is using are alert, log, and pass. The alert function will generate and alert on the system and then log the packet. Log will log the packet without alerting the user. The log files from both these options can be monitored by other programs to take further action. The pass action will allow a packet to be ignored so that false positives are not generated. This will be used to allow the traffic between the UAV and GCS to be passed through without spending system resources monitoring all the packets transmitted between them.

The protocol section for this has the choices of TCP, UDP, IP, and ICMP. During initial testing rules were written to monitor for ICMP packets to detect a system ping. The rules used for the final set up will monitor for UDP protocols since this is the type of packet used by the MAVLink commands.

The src_ip, src_port, dst_ip, dst_port, and direction all serve to specify where the traffic is coming from and going to. The rules were are using are bi-directional, and will detect traffic from all source ip addresses to all destination addresses. This is specified by using ANY for the addresses. Since we have pass rules for the legitimate traffic between the GCS and the UAV the traffic between those two will be ignored. At this point we are not using any custom rule options.

In the final setup, the system will utilized the PulledPork application to keep the rules up to date with the latest network defenses. The Pulled Pork rule files will assign a priority level to the types of attack detected and log all suspicious traffic. If it is noticed that some of the routine traffic between the GSC and the UAV is causing alerts then the local rules files will be edited with pass rules that will allow that traffic through without further analysis.

The log files are monitored by a separate application. The team chose to use Swatchdog to monitor the log files and act on the alerts. The current setup will has Swatchdog run sendip to send a Return to Launch (RTL) signal to the UAV in the event that a Snort alert of priority 2 or higher is found. Swatchdog has the ability to only respond to an attack once in a given period of time to limit the number of alerts, emails, etc. Since the UAV will not be harmed by the RTL command being sent repeatedly it was decided that we will respond to every attack and send the RTL command as often as necessary to ensure recovery of the UAV. These files must be edited everytime the GCS and UAV are paired to ensure that the correct ip address and port are included in the RTL command.

The RTL command was captured by Josh by logging the traffic between the GCS and the UAV while the commands were being sent. By sending the RTL code the drone will be placed into a flight mode that will cause it to fly back to the GPS coordinates that it was initially armed at. This will prevent the loss of the drone.

Since Snort did not have the capability to detect a man in the middle attack in progress, the decision was made to implement an additional application to monitor the ARP map to watch for this type of attack. The ARP map contains a listing of the MAC address and IP address of all devices connected to the network. Arpwatch works by making a copy of that map and then comparing the current version to that original to detect changes that could be caused by arp poisoning attacks or by a man in the middle attack. The Arpwatch program will normally send an email notification of any changes. These alerts are also written to the syslog file. By running a second occurrence of Swatchdog to monitor the syslog file the RTL command can be sent based on those alerts.

Running the virtual machines for the UAV, GCS, and the defensive box locally showed the expected responses when a ping triggered the local rule. The rule for the ping response was then commented out of the configuration files and the modified VM was saved for transfer to the server. Once running on the server it will probably be necessary to update the rules and the response code to reflect different IP addresses.

To allow the sendip to function correctly it was necessary to run it as the root user (sudo). This action normally requires the root password to be entered, but by modifiying the sudoers file, sendip was given permission to run as root without the password. Snorby was implemented to provide a simple method of viewing snort alerts for testing purposes. This application pulls all of the Snort alerts from the log files and displays them in a web-based GUI to allow for easy analysis of the system.

g. Security Onion Vs. SNORT for Intrusion Detection System.

The team explored Security Onion for Intrusion Detection System (IDS) tools. Security Onion was found to be useful for familiarizing oneself to a variety of IDS tools and their functions and operation. However, it also included a number of tools not useful for the goals of this project, and used excessive storage space and memory. Also, Snort requires extensive configuration which made a pre-installed version of it no more useful than a manual install on a clean Linux OS. The decision was made to create a defensive suite built upon a clean version Ubuntu 14.04.3 with Snort as the primary defensive tool that would interface with other tools to provide IDS and IPS.

Snort is a highly effective and popular tool used as an IDS and Intrusion Prevention System. It allows for configurable rule sets to detect attacks and then act upon them. After installation of Snort, the rule sets and configuration file were edited to provide for alerts to the System Log files when an attack was detected. These alerts would then be detected by the tool Swatchdog. Swatchdog monitors the Linux log files, and then performs a pre-determined action when it detects a specified alert. In our case, Swatchdog can be used to detect the text string 'Alert', or the name of the type of the attack that is included in the alert log. Once the text string was detected, Swatchdog executed a custom Bash script to send a command to the UAS. Josh built a VM using these tools and saved it as an OVA file. Tanya continued work on this portion of the project creating an Intrusion Prevention System on the Blue Box.

Kismet is an IDS that is designed to detect traffic over Wi-Fi networks. It is possible to interface Kismet with Snort to provide a more effective coverage of Wi-Fi networks than Snort alone can provide. Snort and Kismet were interfaced to test this configuration and successfully worked together. However, the current stage of the project in simulation only, does not require the Wi-Fi IDS capability. Therefore Kismet will not be used until the project moves to a real world setup.

h. Honey Pot Attack Defense.

The team then began working on the development of a 'Honeypot' defense. This would be an Intrusion Prevention System (IPS) to help deter and confuse attacks. A traditional Honeypot will create a fake serve or a computer on a network for a hacker to detect and being to attack. The honeypot will then often be able record the attempted attacks for later analysis. The Honeypot for this project would need to be similar, but with some unique requirements. Rather than simulating normal computer/server activity, it would need to simulate the activity of a UAV and it's Ground Control Station (GCS). After researching a number of open source and commercial honeypot solutions, it was determined that none were available to simulate the traffic between a UAV/GCS, so the decision was made to create a custom honeypot. The honeypot would serve two roles. Firstly, it would confuse and deter attacks. A hacker observing the traffic between the GCS/UAV would see both legitimate traffic and the honeypot traffic, and would be unable to determine which traffic was genuine without extensively analyzing it. The second purpose of the honeypot is to draw attempted attacks on the honeypot traffic, so that the defensive measures using Snort/Swatchdog can be activated. The more attacks the hacker makes on false traffic, the more opportunities would exist to detect these attacks and apply preemptive defensive measures.

The first suggested architecture of the honeypot was simply to use MavLINK as shown in Kevin's portion of the project to create a false GCS/UAV. This was found to be effective at creating false traffic, however it required additional Virutal Machines and/or actual computers to complete.

A less resource intensive solution was then determined to be creating the GCS/UAV traffic directly. This was accomplished using the SendIP tool. SendIP allows for the creation of arbitrary IP packets over a network. Wireshark was used to analyze the IP traffic created by the actual GCS/UAV traffic. Through careful analyzation of the collected packets it was possible to determine a rough pattern of packets and included data. This pattern was then recreated though SendIP. The goal was not for an exact replication of the packets, as each session between a GCS/UAV would be unique, but simply to approximate the traffic an attacker would expect to see with Wireshark.

The SendIP program includes a looping function. This was used in conjunction with the Linux 'watch' command to create 2 levels of looping with varying times. By adjusting the timing of these loops, different pseudo random patterns of packets could be created to make any attempt at analyzing these packets difficult and time consuming. Multiple concurrent SendIP commands also added to the complexity and randomness, again making it difficult for any attacker to detect the false traffic. After completion of the honeypot script, Zach assisted by encoding it within a Bash script to make it easier to start up and begin sending the code. Blue team also assisted the Red team by capturing and decoding common MavProxy messages so that they could be used in Red Team attacks. For instance, SendIP messages were created that could be used to send Mode Return to Launch, Mode Auto, Mode Guided, Mode Loiter, and Mode Land MAVProxy messages without the need for MavLINK.

VI. System Performance and Testing

Extensive testing of the sub-systems involved in the project were performed in order to ensure these sub-systems operated correctly at both the level of the individual sub-system and the interactions between sub-systems. A full description of the testing plan and a table showing the test results can be seen in Table 1.

First and foremost, testing of the virtualization server itself was performed. This testing included the testing of the PHPVirtualBox web hosted interface (which includes the Apache2 web server, the VirtualBox hypervisor, the password database LDAP directory, and the PHPVirtualBox web interface) to ensure all user accounts could log on to the system and manipulate VMs, first from the local network and then from an external internet host via the RogueNuke.com domain. All tests initiated from the local network were successful from the beginning, however none of the attempts from an external internet connection yielded positive results. Troubleshooting revealed the issue was due to ISP blocking external requests on standard HTTP ports. Initially this problem was solved using a P2PTP-based VPN configuration, but that failed after two days, as the internet-facing IP address for the server was not static. Ultimately the RogueNuke.com domain was re-configured to forward HTTP traffic to the server via non-standard ports combined with a dynamic domain name association. Once this was accomplished, testing yielded successful results from external internet hosts. Server testing also included the testing of the Secure Shell (SSH) server and Remote Desktop Connect (RDC) services. SSH testing involved remote server logon from both internal and external internet hosting via each authorized account and utilizing the command line interface for the VirtualBox hypervisor via this remote connection. RDC testing involved logging on to each VM remotely via the RDC connection, editing a text file on the VM desktop, performing an update to the VM, and successfully shutting down the VM via the RDC connection. Initial testing for both SSH and RDC were successful from the local network, but suffered the same problems identified for the PHPVirtualBox interface noted above. Fortunately, the same solution was successful in resolving these problems; ultimately both the SSH and RDC testing was successful.

Drone VM sub-system and GCS sub-system individual component-level testing included testing of the MAVLink communications protocol configuration. The ability of the ArduPilot software (both ArduCopter and ArduPlane) to receive and respond to MAVLink communications generated locally on the Drone VM was tested to ensure no errors with communications protocol configuration on the Drone VM. Testing was accomplished by sending MAVLink commands to the ArduPilot software which conform to each of the standard commands applicable to both the ArduCopter and ArduPlane variants generated by MAVProxy software installed on the Drone VM. Testing was considered successful when messages generated locally by the ArduPilot software were received and properly decoded by the ArduPilot software. Any instances of lost or misinterpreted MAVLink messages would have constituted test failure, but no instances of this were noted.

Similarly, the MAVLink configuration on the GCS VM was tested. The ability of the GCS software to receive and respond to MAVLink communications generated locally on the GCS VM was tested to ensure no errors with communications protocol configuration on the GCS VM. Testing was accomplished by sending MAVLink commands to the GCS which conform to heartbeat messages and drone status reports generated by ArduPilot software installed on the GCS VM. Testing were considered successful when all messages generated locally by the ArduPilot software are received and properly decoded by the MAVProxy GCS software. Any instances of lost or misinterpreted MAVLink messages would have constituted test failure, but as with the Drone VM, no such failures were observed.

Synchronization between MAVProxy GCS software and Drone Location was also tested at the individual sub-system level. The ability of the Ground Control Station to synchronize GPS location with a simulated GPS location input and display proper map data was tested to ensure the GCS software was capable of detecting and responding to a known location input during system initialization. This was a “go-no go” test. A single initial GPS input was provided via text file generated internal to the GCS. When the GCS initializes with the drone marker in on the map position corresponding to that GPS location the test was considered successful.

Data exchange between flight simulator program and ArduPilot was also tested at the component level. The communication between the simulated PWM signal inputs between the ArduPilot autopilot and the JSBSim flight simulator to provide correct aerodynamic inputs and GPS position information based on simulated engine operation and navigation controllers was examined to ensure the drone performance for a given set of input data conformed to the commanded behavior. Testing was accomplished through 25 trials of all standard commands for both a fixed-wing and quad-copter drone types (ArduPlane and ArduCopter autopilots respectively). Each command applicable to that aircraft was tested during each trial. No performance abnormalities observed during this testing. Parameters such as battery life which do not directly impact response to navigation commands were not tested. Performance in response to environmental conditions, such as wind shear, likewise were not tested.

Upon the completion of individual sub-system testing, the interactions between the GCS and Drone sub-systems were examined. Specifically, MAVLink communication between the Drone VM and GCS VM were tested, the objective of which was to ensure MAVLink messages generated external to the Drone VM and transmitted using the UDP communications protocol were properly received and processed by the ArduPilot software. Simultaneously, the testing ensured MAVLink messages generated by the Drone were properly transmitted and processed by the MAVProxy GCS software on the GCS VM. Testing was accomplished by using the MAVProxy GCS software on the GCS VM to transmit commands to the Drone VM and observe drone appropriate response given those commands. Ten iterations of each command applicable to both the ArduCopter and ArduPlane variants of the were tested. The criteria for a successful test was proper drone response for each individual command with zero losses. Likewise, all status messages sent by the drone must have been received and properly interpreted by the MAVProxy GCS. Finally, for the duration of the testing, the drone position, speed, altitude, and heading must have at all times been synchronized between the Drone and the GCS, as evident by simultaneous observation of drone location and behavior on the map status pages of both the Drone VM and GCS VM. Any observed loss of synchronization would have constituted test failure. However no such failures were noted.

Testing of the intrusion detection system was performed initially to validate that the programs all worked as designed. Local rules were written for Snort which caused an icmp ping to be logged as a priority 1 and then priority 2 alert. A ping was sent to the VM and Snorby was used to verify the correct priority alert was generated. Then the network traffic was viewed to ensure that the RTL command was sent by Swatchdog. All iterations of the testing performed as expected.

Next the IDS system was run with the GSC and UAV to verify that no false positives occured. The UAV and GCS were run to battery shutdown 10 times, providing approximately 300 minutes of flight time. During these flights there were no Snort alerts generated.

Honeypot testing was initially conducted locally with success. The honeypot was able to create streams of data, which when observed through Wireshark, closely matched the actual traffic. The rate of packet creation by the honeypot was found to match the rate of creation of a sample of the actual GCS/UAV traffic with an accuracy of 91.9%. The packet lengths of the honeypot matched the packet lengths of the GCS/UAV with an accuracy of 81.5%. This was deemed to sufficiently simulate actual GCS/UAV traffic and to prove viability of the honeypot. The SendIP tool can also be easily modified to create more or less traffic in the event of a GCS/UAV combination with different characteristics.

The honeypot testing was then attempted using the virtual setup as discussed above. An error was discovered in the results, as Wireshark on a third party system would not detect the SendIP traffic. After some research, it was discovered that the simulation’s reliance on using the eth0 port (simulating a wired connection) would not fully simulate the effects of a WiFi system. In particular, when eth0 is used to transmit SendIP messages, the message leaves the host computer with ‘fake’ IP address and is caught by the router. The router will then reject these messages as it does not know the location of said fake IP addresses. This would not be a factor or problem in a wireless setup, so this error was ignored for our purposes. If running the simulation using eth0, it is recommended to analyze the honeypot data from within the Blue VM in order to observe all expected traffic.

The Red team performed attacks against the GCS and UAV without the defenses to provide a metric that the later results could be compared to. The attacks were ran 100 times against the system. The man in the middle attack was 100% successful and the rebroadcasting attack was 96% successful.

These attacks were repeated with the Blue VM running. Because of the simulated network using a wired configuration, and the Red team attacks already set up against the correct IP addresses, the effectiveness of the honeypot and Snort were not tested. Arpwatch was able to detect 100% of the attacks by the Red team, but the RTL command was not received by the UAV. It was discovered later that the swatch configuration files were not edited to contain the correct port and ip addresses, so the commands may have been sent but were not acknowledged by the UAV. Figs 8, 9, and 10 show some of the arpwatch alerts extracted from the syslog file after the testing was completed. The alert can be seen for when the Red Team VM initially connects to the network in Fig. 8, an ip address change can be seen in Fig. 9, and Fig 10 shows an address 'flip flop'.

```
Apr 29 17:46:14 blue arpwatch: new station 192.168.103.128 08:00:27:23:16:2a eth0
```

Fig. 8 – Red VM connected to network

```
Apr 29 17:46:25 blue arpwatch: changed ethernet address 192.168.103.113 08:00:27:23:16:2a (08:00:27:f9:54:3d) eth0
```

Fig. 9 – Red VM change IP address

```
Apr 29 17:46:36 blue arpwatch: flip flop 192.168.103.113 08:00:27:f9:54:3d (08:00:27:23:16:2a) eth0
```

Fig. 10 – Flip Flop of assigned IP addresses

This final alert was repeated 100s of times throughout the man in the middle attack. If the RTL was sent correctly, the UAV would have returned. The test plan is based on progressive testing of sub-systems independently, testing of the interactions between sub-systems, and finally testing of the complete composite system. For purposes of this testing, sub-systems are identified as the Drone Simulation VM, Ground Control Station Simulation VM, Red Team VM, Blue Team VM, and Virtualization Server. The purposes of testing are to demonstrate both verification and validation of the simulator and all sub-systems therein.

By utilizing a "bottom-up" approach, we can effectively ensure the above sub-systems conform to both the design parameters and functional requirements before integrating them into the composite whole. This process will allow us to identify problems with either verification or validation at the lowest possible level before progressing on to the more complex interactions between sub-systems and finally to the complete system.

VII. Sub-Systems Testing

i) Part I. Sub-system testing

This testing phase tests the operation and performance of vital functions organic to each sub-system independently. During this phase of testing, any failures detected will be addressed by modifying the system under test and repeating testing until no failures occur.

Drone Sub-System Testing. The ability of the ArduPilot software (both ArduCopter and ArduPlane) to receive and respond to MAVLink communications generated locally on the Drone VM will be tested to ensure no errors with communications protocol configuration on the Drone VM. Testing will be accomplished by sending MAVLink commands to the ArduPilot software which conform to each of the standard commands applicable to both the ArduCopter and ArduPlane variants generated by MAVProxy software installed on the Drone VM. Testing will be considered successful if all messages generated locally by the ArduPilot software are received and properly decoded by the ArduPilot software. Any instances of lost or misinterpreted MAVLink messages will constitute test failure.

Ground Control Station Sub-System Testing. The ability of the Ground Control Station to synchronize GPS location with a simulated GPS location input and display proper map data will be tested to ensure the GCS software is capable of detecting and responding to a known location input during system initialization. This is a "go/no-go" test. A single initial GPS input will be provided via text file generated internal to the GCS. If the GCS initializes with the drone marker in on the map position corresponding to that GPS location the test will be considered successful.

The ability of the GCS software to receive and respond to MAVLink communications generated locally on the GCS VM will be tested. Testing will be accomplished by sending MAVLink commands to the GCS which conform to heartbeat messages and drone status reports generated by ArduPilot software installed on the GCS VM. Testing will be considered successful if all messages generated locally by the ArduPilot software are received and properly decoded by the MAVProxy GCS software. Any instances of lost or misinterpreted MAVLink messages will constitute test failure.

Virtualization Server Sub-System. The sub-system for this testing includes the Apache2 Web Server, the PHPVirtualBox interface, the VirtualBox hypervisor, and the password database. Testing is accomplished by successfully logging into the PHPVirtualBox interface, first from a web browser installed on the local network and then from an external internet connection. Testing is successful if each of the user accounts can be logged into and the VMs may be started, stopped, and modified via the PHPVirtualBox web interface.

SSH Server Connectivity testing examines the ability to remotely log into the server operating system via a Secure Shell (SSH) connection. Testing is considered successful if the sever can be logged on to successfully from each account. Testing will include the command-line interface to start, stop, reconfigure, and export VMs through VirtualBox, and the navigation and editing of flies related to the project on the server, as demonstrated by the ability to edit the PHPVirtualBox configuration file.

Remote Desktop Connect testing examines the ability to remotely operate VMs using the Microsoft Remote Desktop Connect software. Testing will be accomplished by establishing a RDC connection to each VM, logging in successfully to the VM via the RDC connection, editing a text file on the VM desktop, performing system updates via RDC, and successfully shutting down the VM via the RDC connection. Additionally, testing will not be considered successful unless these functions are able to be accomplished from both a computer on the local network and from an external internet connection via the RogueNuke.com domain.

Blue (SNORT and HoneyPot) Sub-System. The SNORT sub-system will be tested by running the box and sending test attacks that will be recognized as the different priority levels to ensure the correct responses are sent. Testing of this phase will be successful if the system send the correct swatchdog response. The test will be considered successful if the expected response to all of the attacks is executed.

Testing for the 'HoneyPot' will be conducted using WireShark, Wireshark's statistical analysis tool, and Minitab. The HoneyPot creates traffic that appears to come from multiple UAV's and GCS's using the SendIP tool. This traffic can be captured by WireShark, and then compared to actual UAV and GCS traffic. Resident within WireShark are statistical tools that allow for the ability to determine a number of properties of this captured data, such as packet length and frequency of traffic from individual IP's. A comparison in amount of traffic will be conducted using Minitab, with a statistical analysis resulting in percentage of 'HoneyPot' traffic compared to actual traffic. A graphical and numerical comparison of packet length will be conducted using data from 'WireShark' analyzed by Minitab. The 'HoneyPot' will be considered successful if it can create multiple false UAV/GCS's that generate 90-110% of the number of packets as the actual UAV/GCS, and if the false traffic packet length has a similar (90%) normal distribution to the actual.

ii. Part II. Sub-system Interactions Testing

This phase of testing tests the performance of key seams and interactions between individual sub-systems. The focus is on the response of individual sub-systems in response to inputs into the sub-system and the resulting outputs generated by the sub-system.

Drone Sub-System Testing / GCS Sub-System Testing will be accomplished by using the MAVProxy GCS software on the GCS VM to transmit commands to the Drone VM and observe drone response is correct given those commands. Ten iterations of each applicable command will be tested. The criteria for a successful test is proper drone response for each individual command with zero losses. Likewise, all status messages sent by the drone must be received and properly interpreted by the MAVProxy GCS. Finally, for the duration of the testing, the drone position, speed, altitude, and heading must remain synchronized between the Drone and the GCS. Any observed loss of synchronization will constitute test failure.

Red (Kali Linux) Sub-System / Drone VM Sub-System. To properly demonstrate the interaction between the Drone VM and the Red Sub-System ARPSpoofing is used. This will demonstrate the ability to read traffic being sent back and forth between the Drone VM and GCS VM, identifying key elements such as the Drone's IP address as well as its MAC address. This will show the interaction between all three sub-systems without actually attacking and a key process in what happens in a real-world situation.

Blue (SNORT) Sub-System / GCS Sub-System Testing / Drone VM Sub-System. The snort sub-system will be tested with the GCS and the Drone operating to ensure the rules do not generate false positives under normal operation. During this phase a 30 minute mission will be flown 10 times and the number of priority one and priority two alerts will be monitored. Any false alerts will constitute a failure and will require modification of the rules files.

iii. Part III: Composite System Testing

This final phase of testing examines the performance of the composite system, during both unprotected and protected attacks against an operating drone and GCS during routing commands.

Undefended Composite Attack Testing. This testing is accomplished without the defensive (Blue) VM in operation to establish a baseline of performance for an undefended system. 100 iterations of both a man-in-the-middle attack and a rebroadcasting attack against the operating Drone and GCS VMs and the success rate of these attacks will be examined. For the man-in-the-middle attack, an ARP poisoning attack using ARPSpoof will be used to establish the man-in-the-middle scenario. From there WireShark will be used to identify the Drone and GCS IP addresses and ports.
Finally, MAVProxy/MAVLink will be used to take control of the drone. An attack is considered successful if the attacker is able to issue a command to the drone to which the drone responds. For the rebroadcasting attack, commands issued by the GCS to the Drone will be captured by the attacker using WireShark and then rebroadcast at a later time using Sendip. An attack is considered successful if the drone responds to the rebroadcast command. Finally, the LowOrbitIonCannon (LOIC) will be used to generate a denial of service attack. The attack is considered successful if the LOIC successfully swamps the network router. Drone response to the LOIC attack will be observed and the time required to re-establish communications between the GCS and the Drone once the attack is ceased will be measured.

Intrusion Detection Composite Testing. Testing of the entire system will be accomplished by initializing the UAV, GCS, Snort system and honeypot virtual machines and then launching the red team VM. The tests from the undefended attack testing will be repeated and the rate of success from these attacks will be compared to the testing results without the defensive packages running and will be used to determine the effectiveness of the defensive software. Table 1 show the system results of sub-systems of the project with a testing metric of pass or fail status.

Table 1. System Results

Test	Goal	Pass /Fail
	Individual Sub-Systems	
Drone	Target - All messages received and decoded	
	Actual - All messages received and decoded	Pass
GCS - GPS Testing	Target - Drone marker initializes in correct position	
	Actual - Drone marker initializes in correct position	Pass
GCS - MavLink Communications	Target - All messages received and decoded	
	Actual - All messages received and decoded	Pass
Server - User Account Access	Target - All users able to log in and access VMs	
	Actual - Using document put out by Kevin all users can log in	Pass
Server - SSH Connectivity	Target - All accounts can log in through SSH connection	
	Actual - Accounts can log in through SSH connection.	Pass
Server - Remote Desktop Connection	Target - Access and use of all VMs successful using RDC	
	Actual - Access and use successful but slow	Pass
IDS box	Target - Correct response to all priority attacks	
	Actual - Correct response sent to all priority attacks	Pass
HoneyPot - Number of Packets/second	Target - 90-110% of actual UAV/GCS packet rate	
	Actual - 1-(HP 11.94-UAV 11.04)/UAV 11.04=92.44%	Pass
HoneyPot - Packet Length	Target - 80% or higher of normal distribution to actual packets	
	Actual - 1-(UAV 21.0 -HP 17.14)/UAV 21.03=81.5%	Pass
	Sub-system Interactions	
Drone - GCS Command response	Target - Correct responses and zero command losses	
	Actual - Correct responses and zero command losses	Pass
Drone - GCS Synchronization	Target - No observable loss of synchronization	
	Actual - Synchronization losses occur if left idle for too long	Pass
Red box - Drone	Target - Able to read traffic and identify IP and MAC address	
	Actual - Able to read traffic and identify IP and MAC address	Pass
IDS - GCS - Drone	Target - No false positives during normal drone operation	
	Actual - No false positives of any alert level	Pass
	Composite System	
Undefended Rebroadcast	Target - none, used to establish baseline	
	Actual - 96%	
Undefended Man-in-the-Middle	Target - none, used to establish baseline	
	Actual - 100%	
Defended Rebroadcast	Target - Detect and respond to all attacks	
	Actual - 0% undetected - NO RTL command received by UAV	FAIL
Defended Man-in-the-Middle	Target - Detect and respond to all attacks	
	Actual - 0% undetected - NO RTL command received by UAV	FAIL

viii. Ethical Considerations

Any research into possible malicious exploits of security vulnerabilities in software systems carries with it an inherent ethical concern for how that research may be used. Guidelines for this type of research are provided in the Federal Information Security Management Act (FISMA) and assorted industry publications. In general, it is the responsibility of an ethical penetration tester to inform manufacturers of security vulnerabilities identified in their products and allow them sufficient time to address these vulnerabilities—should they choose to do so—before disclosing vulnerabilities to general public.

With respect to this project, no new vulnerabilities were identified in the hardware or software being examined. Some of the attacks conducted as part of this project may be novel in the detailed mechanisms by which they are employed, but the vulnerabilities themselves are not new and have been well documented in academic research publications and presentations at various cyber security conferences. Therefore, nothing presented in this paper poses an increased risk to the public good, but demonstration of these vulnerabilities as presented in this project, as well as potential novel solutions, will hopefully further build on existing efforts and encourage industry and academia to respond to these security issues.

Additionally, as one of the primary goals of this research was to create a simulation environment that could be used to test offensive and defensive cyber security techniques without fear of creating actual hazards to the public, the product developed by this project serves to provide a safe and effective mechanism for accomplishing ethical testing without potential harm to existing networks and systems.

ix. Performance Metrics

Metrics for use in evaluation of this project included are: percentage of packet loss between the Drone and the Ground Control Station, percentage of successful Drone mission accomplishment, percentage of attacks/intrusions detected, mean time before attack/instruction detected, percentage of successful Drone return to start, percentage of malicious commands executed by the Drone, and frequency of unrecoverable Drone error.

x. CONCLUSION

Through the use of open-source Linux-based virtual machines, a fully customizable UAV simulation environment can be created for both software-in-the-loop and hardware-in-the-loop cyber security testing of fixed and rotory-wing ArduPilot over MavLink UAVs at a fraction of the cost required for testing with physical UAV assets. This project demonstrated how both offensive and defensive cyber security operations could be tested within such an environment. Utilizing the Kali Linux software suite, man-in-the-middle attacks and rebroadcasting attacks had 100% and 96% success rates respectively against a stock ArduPilot UAV; these results are consistent with those found in the literature for physical UAVs. Furthermore, the simulation environment developed in this project was used to design, develop, and test a novel UAV security scheme based on an automated SNORT-based intrusion detection system external to both the UAV and the ground control station. Initial tests of the individual systems have been successful and final testing of all areas is in process. The simulation environment developed in this project is effective for both evaluating the potential vulnerability of existing UAV control and communications systems, as well as for developing and testing potential solutions to those vulnerabilities.

Acknowledgement

This project is sponsored and made possible by Rockwell Collins Charitable Grant Allocation and Wells Fargo Foundation grants in 2015-2016.

REFERENCES

Buchanan, C., & Ramachandran, V. (2015). *Kali Linux: Wireless Penetration Testing Beginner's Guide*. Birmigham, UK: Packt.

Javaid, A. Y., Sun, W., Devabhaktuni, V. K., & Alam, M. (2012). Cyber security threat analysis and modeling of an unmanned aerial vehicle system. *2012 IEEE Conference on Technologies for Homeland Security (HST)*, 585–590. https://doi.org/10.1109/THS.2012.6459914

K, H. (2015). Todd Humphreys' Research Team Demonstrates First Successful GPS Spoof. Retrieved from https://www.ae.utexas.edu/news/features/todd-humphreys-research-team-demonstrates-first-successful-gps-spoofing-of-uav

Kim, A., Wampler, B., Goppert, J., Hwang, I., & Aldridge, H. (2012). Cyber Attack Vulnerabilities Analysis for Unmanned Aerial Vehicles. *AIAA Infotech at Aerospace 2012*, (June), 1–30. https://doi.org/10.2514/6.2012-2438

Lakhani, A., & Muniz, J. (2013). *Web Penetration Testing with Kali Linux*. Packt Publishing.

MAVLink Micro Air Vehicle Communication Protocol – QGroundControl GCS. (2015). Retrieved from http://qgroundcontrol.org/mavlink/star

P, B. (2010). About Scapy. Retrieved from http://www.secdev.org/projects/scapy/doc/introduction.html#about-scapy

Paganini, P. (2013). Hacking Drones - Overview of the Main Threats. Retrieved from http://resources.infosecinstitute.com/hacking-drones-overview-of-the-main-threats

S, F. (2003). Detection Tools: Snort. Retrieved from http://www.informit.com/guides/content.aspx?g=security&seqNum=48

JOURNAL OF UNMANNED AERIAL SYSTEMS

PEER-REVIEWED ARTICLE

OPERATIONAL ORGANIZATION OF SMALL UNMANNED AERIAL SYSTEM PHYSICAL AIRSPACE

Dr. Daiheng Ni
Associate Professor of Civil Engineering at the University of Massachusetts Amherst

Mr. Jared Geller
Undergraduate Student in Department of Civil and Environmental Engineering at the University of Massachusetts Amherst

ABSTRACT

Surging demand for public and civil applications of small unmanned aerial systems (sUAS) has urged the government, the industry, and the academia to explore ways to integrate such vehicles into already crowded national airspace. With research actively on-going at the high level which concerns about policies and regulations and at the low level which addresses sense and avoid, A wide gap has been identified in between. Hence, this paper aims at the middle-level sUAS operational organization problem. The objective of this paper is to integrate ideas presented by the FAA, NASA, Amazon, and academia regarding airspace organization with successful experiences of traffic operation in other modes of transportation in order to design safe and efficient operation modes for sUAS traffic.

Keywords: small unmanned aerial systems (sUAS), traffic operation, airspace design, modes of operation

Introduction

Due to surging demand for public and civil applications of small unmanned aerial systems (sUAS), which are unmanned aerial vehicles (drones) that weigh less than 55 pounds, academia and government agencies have been extensively studying how the United States could incorporate such vehicles into an already crowded airspace. In studying the methods to control the proliferation of sUAS, there are three different fields of study to consider. Firstly, there is the high level focus, which includes topics such as sUAS policies, regulations, and traffic management. This arena has already been extensively studied by NASA, which has presented a concept of operations for unmanned aerial system traffic management (UTM), by the FAA, which has proposed rules and policies for sUAS, and by Amazon, which has created a rudimentary physical airspace design (NASA, 2015) (De Los Santos & Rios, 2015) (Dillingham, 2015) (Golson, 2015). Secondly, there is the low level focus, which includes modeling interactions between sUAS that are equipped with sense-and-avoid (also referred to as detect-and-avoid or detect, sense, and avoid) technology. Numerous academic papers have attempted to address definitions of well-clear as well as how computer algorithms would separate aircraft, both of which are crucial in developing sense-and-avoid for sUAS (Munoz, Narkawicz, Chamberlain, Consiglio, & Upchurch, 2015) (Hottman, Hanson, & Berry, 2015).

However, there is a wide gap in between the high level and low level arenas: the mid-level focus. Very few have considered the mid-level focus, which is meant to address how sUAS traffic is organized and separated within the airspace (NASA's unmanned traffic management concept does not address this in great detail). The motivation behind this paper is to fill this gap in order to make upcoming low-altitude sUAS operations safer and more efficient. Our vision is to allow high-speed, beyond-visual-line-of-sight sUAS traffic to operate seamlessly within an airspace which includes many different vehicles. The objective of this paper is to integrate ideas presented by the FAA, NASA, Amazon, and academia regarding airspace organization with successful experiences of traffic operation in other modes of transportation in order to design safe and efficient operation modes for sUAS traffic. In order to achieve this, our approach will be 1) to incorporate ideas and successful experiences from other modes of transportation, and 2) to consider realistic designs which take privacy and safety concerns into serious consideration.

Recent sUAS Developments

While not much literature currently exists about organizing the airspace for small unmanned aerial systems (sUAS), which are best described as drones weighing less than 55 pounds, two entities have taken a keen interest in the topic. Firstly, NASA has delved into the subject out of necessity. In 2013 and 2014, 128,000 and 430,000 sUAS were sold in the U.S., respectively. 2015 is on pace to reach over 700,000 drone sales (CBSNews, 2015). Several incidents have occurred where sUAS, currently under regulated, have interfered with manned aircraft operations. Recently, hobbyists flying sUAS over California's wildfires have prevented firefighters from flying helicopters and planes over certain areas to drop water and flame retardant. The U.S. Forest Service has recorded 13 wildfires in 2015 in which sUAS interfered with manned aerial firefighting operations, 11 of which occurred from late June through the end of July (CBSNews, 2015). There are also reports of sUAS causing emergency landings for both recreational and commercial aircraft. On August 2, 2015, a commercial airline pilot approaching New York's JFK Airport reported a sUAS in the vicinity of the aircraft. On August 9, only a few days later, 4 commercial airline pilots reported a drone on approach to Newark International Airport while flying between 2000 and 3000 feet (CBSNews, 2015). Clearly, the government needs help establishing laws, rules, and regulations that restrict drones to safe flying environments while not being too overbearing. While the FAA is focused on administration and public policy, NASA is focused on the technical elements of organizing sUAS airspace. NASA has already shared its plan for airspace organization, which it calls unmanned aerial system traffic management (UTM). UTM, according to NASA, would allow for "safe and efficient low-altitude airspace operations by providing services such as airspace design, corridors, dynamic geo-fencing, severe weather and wind avoidance, congestion management, terrain avoidance, route planning and re-routing, separation management, sequencing and spacing, and contingency management (De Los Santos & Rios, 2015)."

Secondly, Amazon has also put some effort into airspace organization, motivated by the potential for reduced delivery costs and increased profits. The company plans on using sUAS through its *Amazon Prime* service to allow for package deliveries that take less than one day. This service is currently referred to as *Prime Air*. In spite of Amazon's profit-driven and aggressive push to allow for sUAS package deliveries, the company's engineers and researchers have theorized and presented their own well-thought-out plan for airspace organization which builds upon ideas presented by NASA. Amazon's plan includes altitude stratification, otherwise known as separating the airspace into vertical layers for different aircraft. From ground-level to 200 feet, "low-speed localized traffic" would operate. This would be the zone where package delivery drones would ascend and descend to make the actual deliveries. This is also where hobbyists, photographers, and other operators involved in low-speed, low-altitude flights would use their vehicles. From 200 to 400 feet, "high-speed transit" vehicles would operate. This would include package delivery and emergency vehicles which seek to reach a destination as quickly as possible. Airspace between 400 and 500 feet would be a no-fly zone, used to provide a buffer between unmanned and manned aircraft, which are required to be above 500 feet for the plan to function properly. Importantly, drones would not be able to fly within a certain radius of airports. *Figure 1* includes Amazon's diagram of the plan (Golson, 2015).

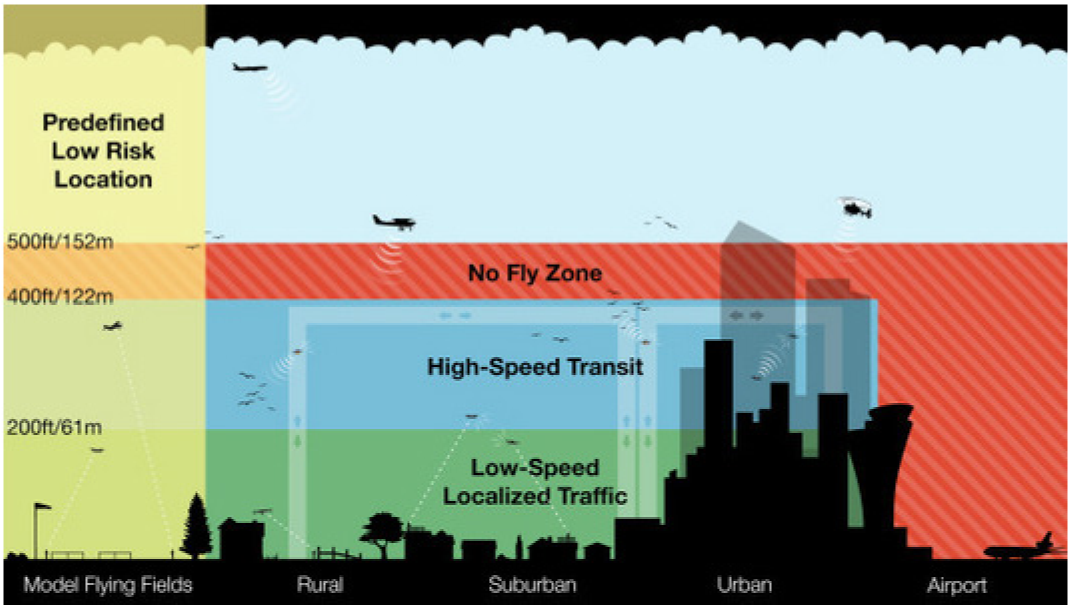


Figure 1: Amazon's Proposed sUAS Airspace Organization

Organizational Methods

3.1 Overview

In order to design an airspace model for sUAS, it is necessary to have an understanding of what objectives sUAS have. Firstly, there are sUAS which need to reach a destination as efficiently as possible, most likely to make a delivery. These high-speed, beyond visual-line-of-sight sUAS will be referred to as **destination sUAS**. Other sUAS will operate sporadically within a relatively small volume of airspace, such as those used for photography, agriculture, infrastructure monitoring, and more. These will be referred to as **proximity sUAS**. Some sUAS might even fall under both categories.

Every owned parcel of land is expected to have exclusive air rights within the land's property lines and up to 83 feet, barring certain exceptions (Henn, 2014). This will be known as **private airspace**. Property owners will be able to operate proximity sUAS as they please within the volume that is their private airspace. Local regulations will dictate how closely proximity sUAS operating within private airspace can approach the boundaries of other properties or private airspace. Properties which include tall buildings will have **tailored airspace**, private airspace with adjusted boundaries, to prevent other sUAS from operating too closely. Property owners can waive their **exclusive right** to their private airspace for any period of time, and this ability will mostly be used to allow package deliveries.

Above altitudes of 83 feet, marking the point at which private airspace tops out, is **free airspace**. Free airspace will also exist at points below 83 feet in portions of airspace above public property, and local regulations will dictate how far below 83 feet sUAS pilots can operate over public land, if at all. Any free airspace below 83 feet and above public land will be known as **extended free airspace**. Only operators with permission from some government overseer of sUAS operations will be able to enter free airspace from private airspace, which would most likely consist of package delivery companies or other destination sUAS (NASA, 2015) (De Los Santos & Rios, 2015). The speed limit in free airspace is expected to be low due to the freedom of operation allowed in this sector. In addition, sense-and-avoid technology would be required.

Some operators may request **reserved airspace** from the overseer of sUAS operations (NASA, 2015) (De Los Santos & Rios, 2015). Reserved airspace consists of some 3-D volume of free airspace allocated exclusively to a certain operator for some period of time, so long as other operators would not be adversely affected by not being able to operate in that particular area. **Temporarily reserved airspace** would be appropriate for operators who need access to a certain part of free airspace for proximity sUAS operations for periods such as an hour or day. **Permanently reserved airspace** would be appropriate for airspace above properties which disperse many destination sUAS, such as package delivery hubs. The rights for permanently reserved airspace would likely need to be renewed after certain intervals of time, such as a year, to prevent "senior rights" operators from dominating the airspace.

In areas of high destination sUAS traffic, such as the airspace between a package delivery hub and a large suburb, **corridors** could be set up to shuttle sUAS safely and efficiently back and forth. Corridors would be high-speed virtual tubes which sUAS could merge in and out of so long as they follow safety protocols. Speed limits would be higher to encourage operators to utilize the safer corridors instead of navigating through free airspace for the entire duration of the operation. *Figure 2* includes a visual overview of this airspace organization. As mentioned earlier, the delivery company operating the sUAS delivery hub should have the ability to reserve airspace permanently so that delivery sUAS can reach high-speed corridors from the delivery hub's preordained private airspace (all properties have automatic air rights up until a certain height). Corridors will exist to funnel traffic safely and efficiently through high traffic areas, usually between delivery hubs and suburbs and cities. Vehicles may exit corridors as they please into free airspace, or they can travel to the end of a corridor, which empties into free airspace. Within free airspace, other operators can reserve airspace for so-called proximity sUAS operations, which are those that require a 3-D area for spontaneous yet leisurely motions, such as photography or inspections.

Private residences will most likely have air rights up until 83 feet, and in order to receive deliveries, property owners must waive their exclusive access to their own airspace. Taller structures, especially infrastructure which requires monitoring for deficiencies, such as a water tower, will have private airspace tailored to a non-standard area to keep other aircraft away and to uphold safety and privacy. Over land that is not private, local ordinances will determine how low vehicles can fly. sUAS are expected to take off and land from commercial properties, like farms, delivery hubs, third-party headquarters, etc., or government-owned sUAS headquarters, all of which will be allowed to operate in their own airspace or reserve airspace in free airspace if necessary. Again, all of this is presented in *Figure 2* below.

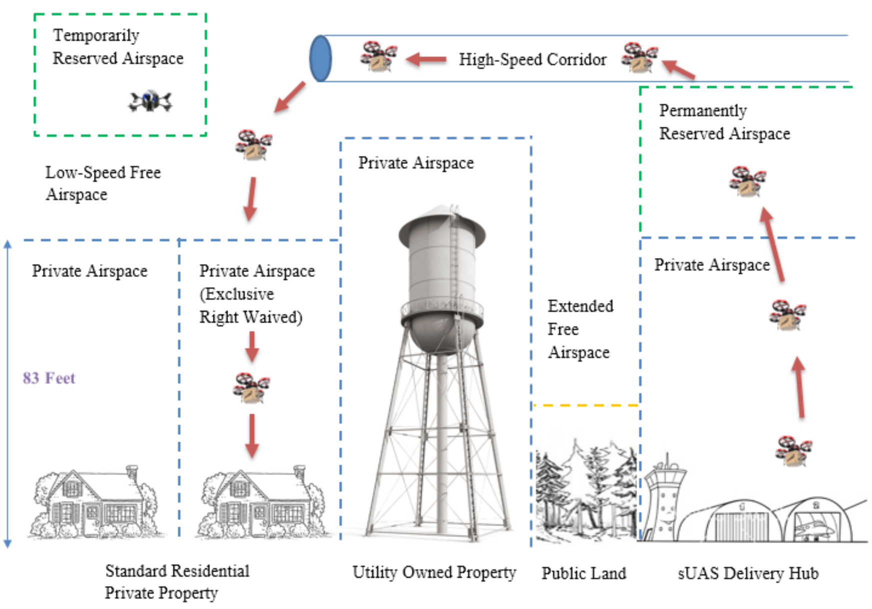


Figure 2: Airspace Organization Overview

3.2 Private Airspace

Airspace immediately above commercial or residential properties is private, since property owners are guaranteed air rights. However, property owners should be able to waive their airspace rights to other operators, whether temporarily, indefinitely, or permanently. Emergency vehicles would likely be exempt from requiring special permission to operate in private airspace, though this will depend on local ordinances. In addition, how high air rights stretch above a property before bordering free airspace needs to be determined by regulators, whether on the local, state, or federal levels. Based on a Supreme Court case dating back to World War II, the accepted number for the minimum elevation for which aircraft should fly over private property is 83 feet (Henn, 2014). The lateral boundaries will not be vertical extensions of the property lines, but rather a smaller outline of the property. The distance by which the perimeter is reduced for private airspace will likely be determined by local ordinances as to protect the privacy of adjacent property owners. Based on these boundaries, we can assume that free airspace exists from 83 feet up until 400 or 500 feet, where the cap for sUAS is expected to be, and that any airspace below 83 feet and above private property is private airspace. Other land, such as public land, would be subject to further ordinances. In some cases, the height restrictions above public land might be lower than 83 feet based on local ordinances, and portions less than 83 feet would be known as extended free airspace. *Figure 3* portrays the boundaries between private and free airspace as well as a portion of extended free airspace, which would exist above public land.

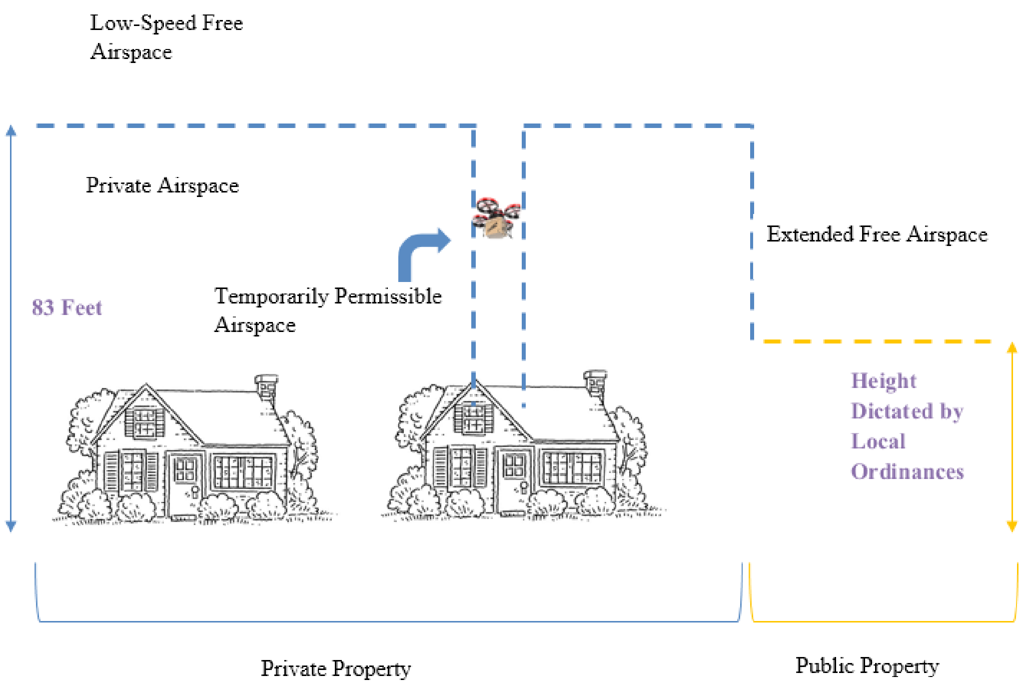


Figure 3: Private Airspace Boundaries and Extended Free Airspace

3.3 Free Airspace

As briefly stated before, free airspace will be open to all sUAS, though speed limits will be lower than in corridors to account for the unpredictable nature of the trajectories of vehicles in this portion of airspace. Free airspace is a transitional zone between corridors and private airspace. Again, private airspace is where sUAS will land and take off from, so long as the private airspace is above the property owned by or affiliated with the sUAS operator. Based on the randomness of this sector, sUAS operators who wish to operate in free airspace must have their vehicles equipped with sense-and-avoid technology. Operators and vehicles which operate here also should be licensed and registered, respectively. In addition, free airspace can be reserved for some period of time, so long as the operational area does not negatively impact other operators or those on the ground. Above residential properties free airspace is expected to begin at 83 feet. Local ordinances will dictate how far extended free airspace will exist below the 83 foot mark above public lands. For buildings or structures taller than 83 feet, free airspace will be tailored around the private airspace required to keep unwanted vehicles away from such structures both horizontally and vertically. *Figure 4* shows what such a tailored free airspace would look like to protect the air rights of high-rises.

JOURNAL OF UNMANNED AERIAL SYSTEMS

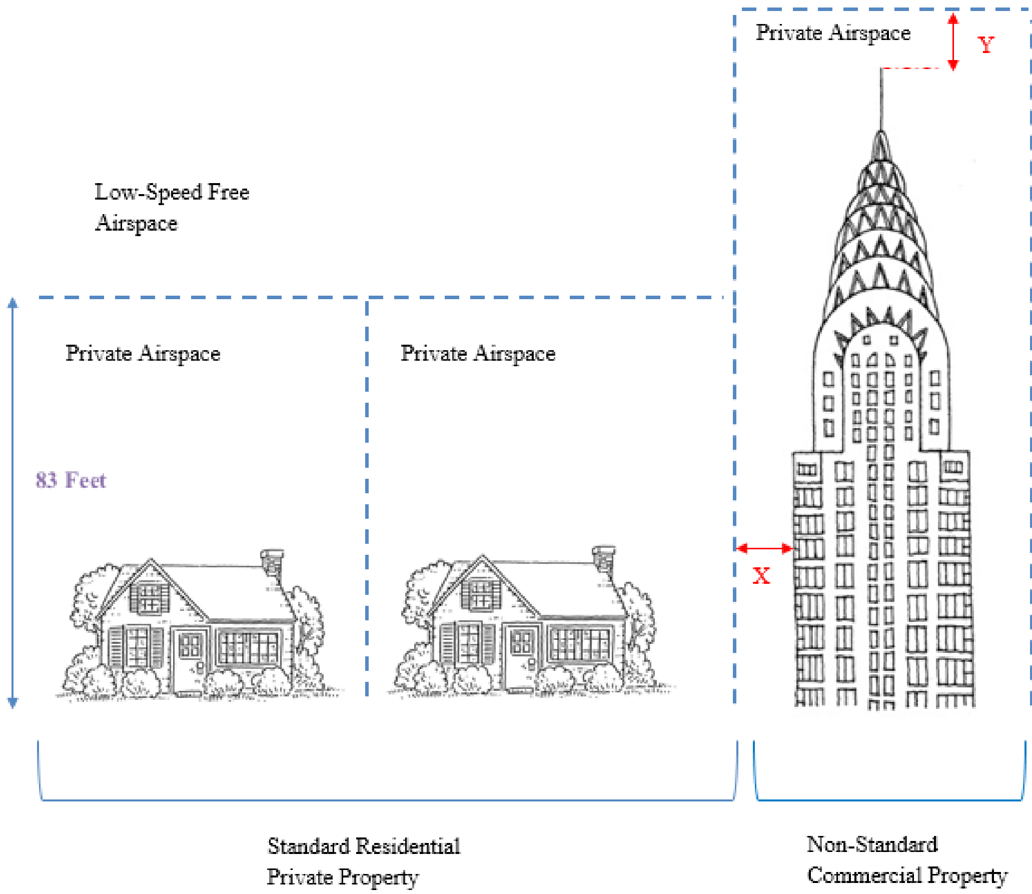


Figure 4: Tailored Free Airspace for Nonstandard Properties

3.4 Reserved Airspace

Reserved airspace would be beneficial for proximity sUAS, those that require a 3-D area for hovering along a slow and unknown trajectory (for photography, inspections, etc.). If companies need to reserve airspace outside of their own boundaries that will either benefit or have little effect on the public or other users, or if public operators or third parties hired by government need to utilize free airspace for some purpose, they should be able to reserve free airspace for the duration which their operation would require. This time span could be anywhere from one day to permanently or indefinitely. The overseer of the airspace would decide which applications for reserved airspace are necessary or unbothersome, and which ones would impair safety or privacy. *Figure 5* shows how reserved airspace could be implemented. Suppose that a local fire department wants to survey a nearby forest for brush fires. If the forest is public land, the department would legally be allowed to use their sUAS up until a height specified by local ordinances, such as 40 feet. Airspace above that would be free airspace. However, the team might need to hover well above the treetops at 70 feet to see signs of smoke. In order to ensure that no other sUAS interferes with this operation, the fire department can reserve the airspace from the overseer of the airspace to conduct the mission.

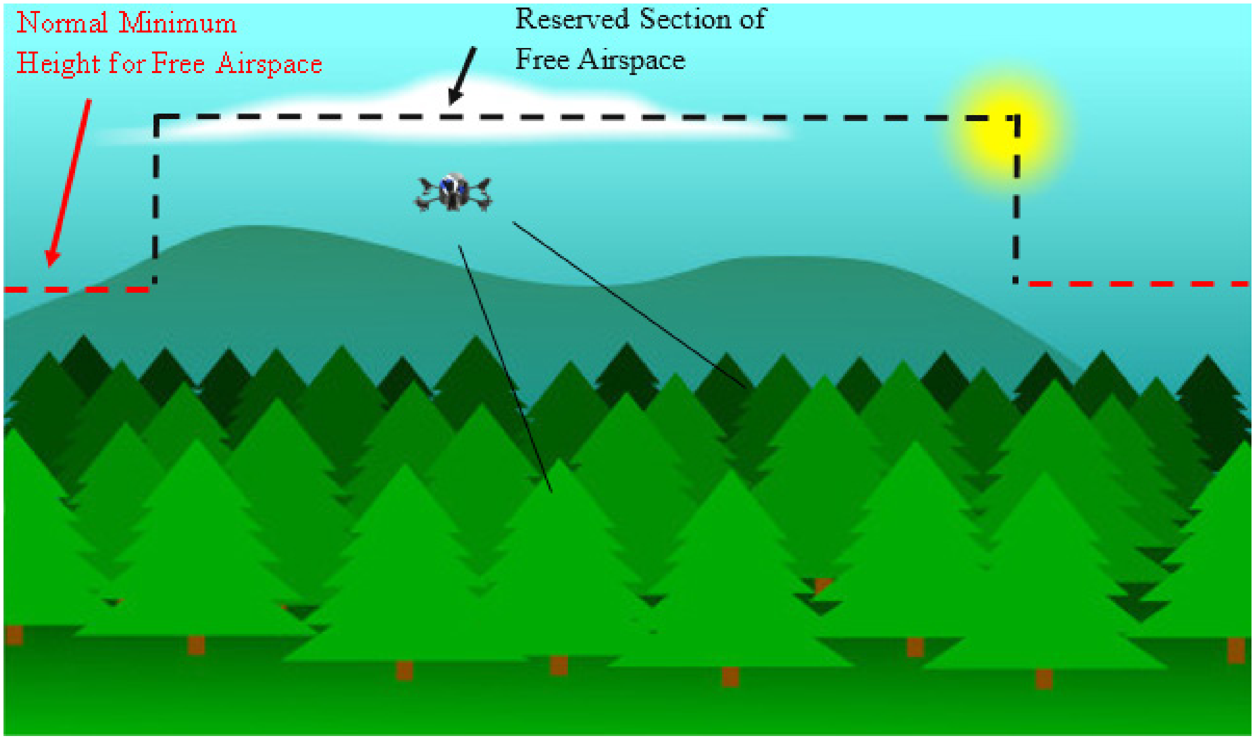


Figure 5: Reserved Airspace Example

3.5 Destination sUAS and Corridors ("Air Highways")

Package delivery sUAS are expected to be high-speed, beyond-visual-line-of-sight vehicles (destination sUAS). As such, they need to be separated from other types of sUAS used for applications which are not meant to allow the vehicle to move from an origin to a destination as quickly as possible (proximity sUAS). In addition, corridors might be necessary before the proper wide-spread autonomous vehicle-to-vehicle separation technologies allow vehicles to fly quickly and freely in any direction while avoiding one another, like ideal gas particles. Therefore, air corridors will provide a safe 3-D pathway for delivery vehicles in relatively high traffic air space.

To describe how air corridors could be useful, we can begin with an illustrative example of a package delivery. A customer first orders a product online and expects that product to be delivered to his or her front door within the next 24 hours. That package must travel from a distribution center or delivery hub. A vertical cylinder could be geo-fenced around the delivery hub to give ascending and descending delivery vehicles exclusive access to that airspace. Up until 83 feet, this would be a given, but in the portion of free airspace between the private airspace and the corridor, the company would have to reserve exclusive air rights. At an altitude high enough to not interfere with low-altitude sUAS operations, but low enough to be well-away from manned aircraft operations, this cylinder could intersect with a corridor. This geo-fenced cylinder could also be an upside-down cone so that delivery drones would not have to ascend or descend in a perfectly vertical manner. *Figure 6* includes a possible rendering for how the airspace could be designed above a package delivery hub.

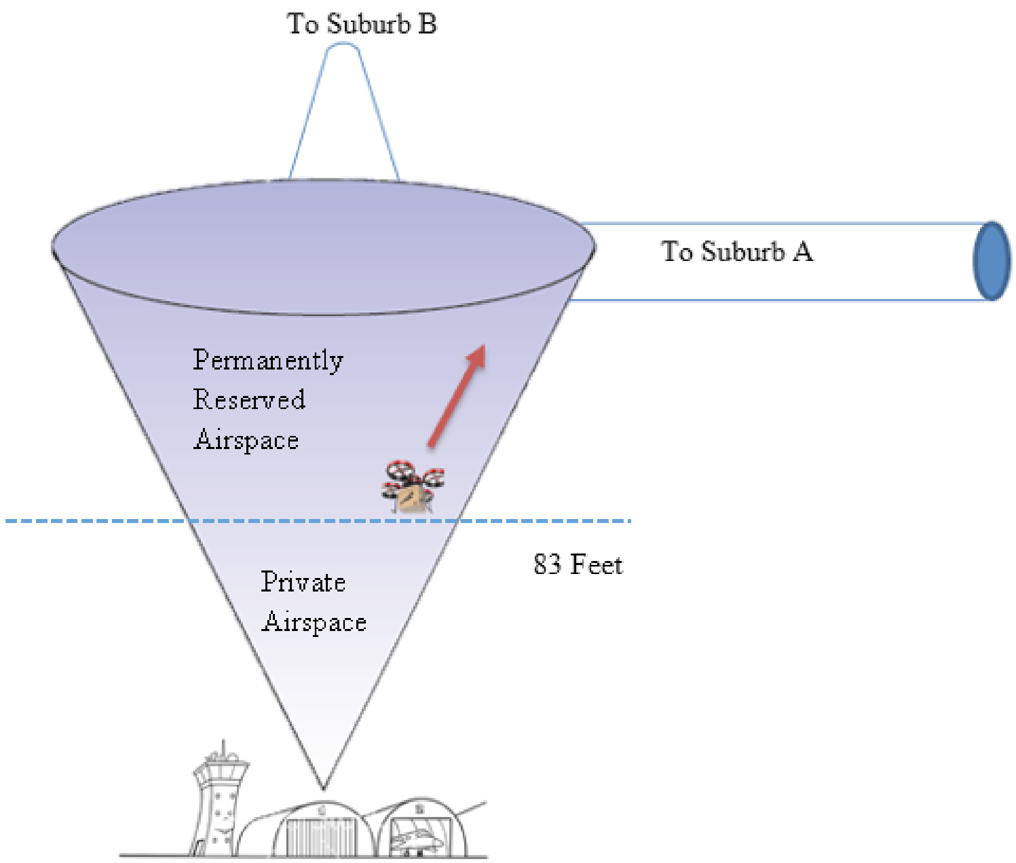


Figure 6: Airspace Diagram for Package Delivery Hub

Corridors could link the airspace above the delivery hub with suburban areas. Once the delivery drone reaches a section of the corridor near the destination, it could exit the corridor and enter free airspace, where maximum allowed speeds are lower due to the spontaneity of vehicles in this portion of airspace. Once above the correct property, the delivery drone could descend vertically towards the property of the recipient of the package. A vertical column above the property would be set up to allow temporary access to reach the recipient's property from free airspace. Once the package is delivered, the delivery drone would need to ascend vertically back to free airspace, where it can then move horizontally and vertically back to the corridor to return to the delivery hub. The reason for vertical take-offs and landings is to protect property rights and the privacy of those not involved in the delivery. *Figure 7* illustrates how package delivery drones would reach their destinations from corridors.

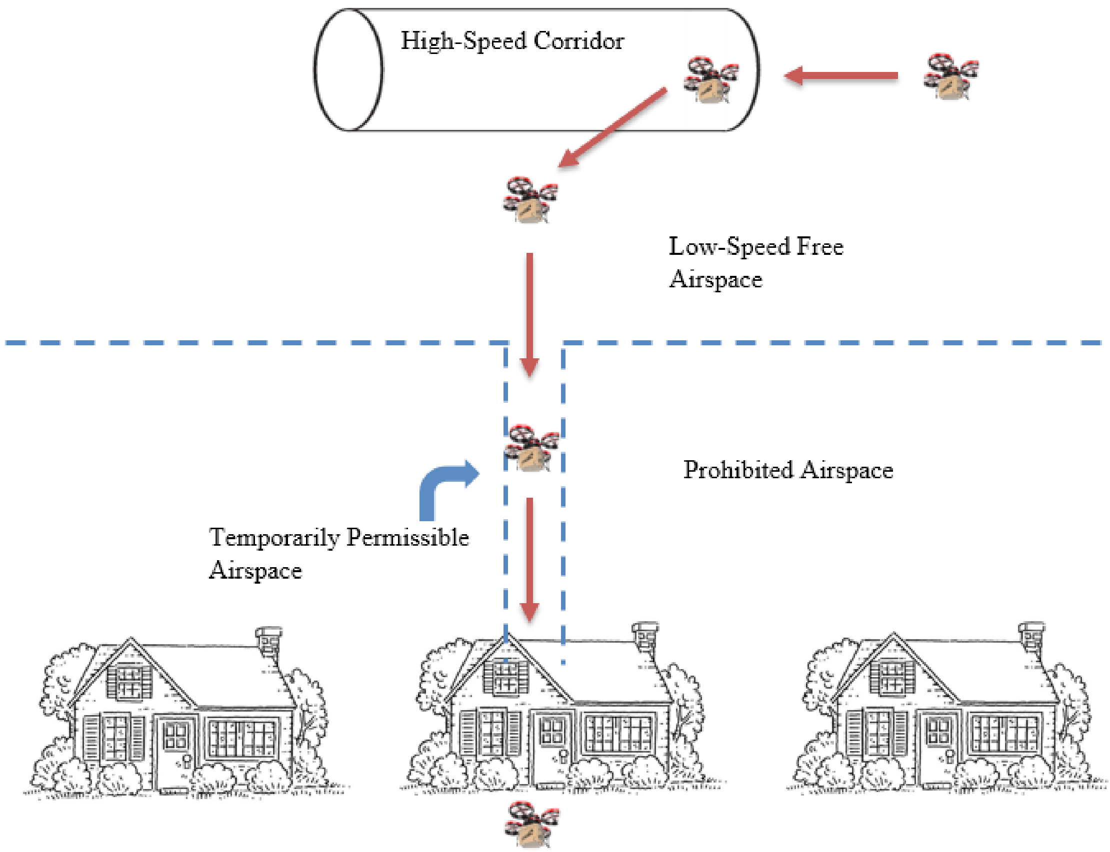


Figure 7: From Corridor to Delivery

One issue that arises from such a setup includes drones safely emerging from and entering high-speed corridors. One solution is to include a set outer radius for merging into corridors from low-speed free airspace or for slowing to appropriate speeds for free airspace from corridors. These would function similar to highway on-ramps and off-ramps, which allow vehicles to reach highways speeds safely before entering and to safely slow to local traffic speeds from highway speeds, respectively. *Figure 8* shows a cross-sectional area of a corridor which helps to illustrate how corridors, merging and exit zones, and free airspace could be designed.

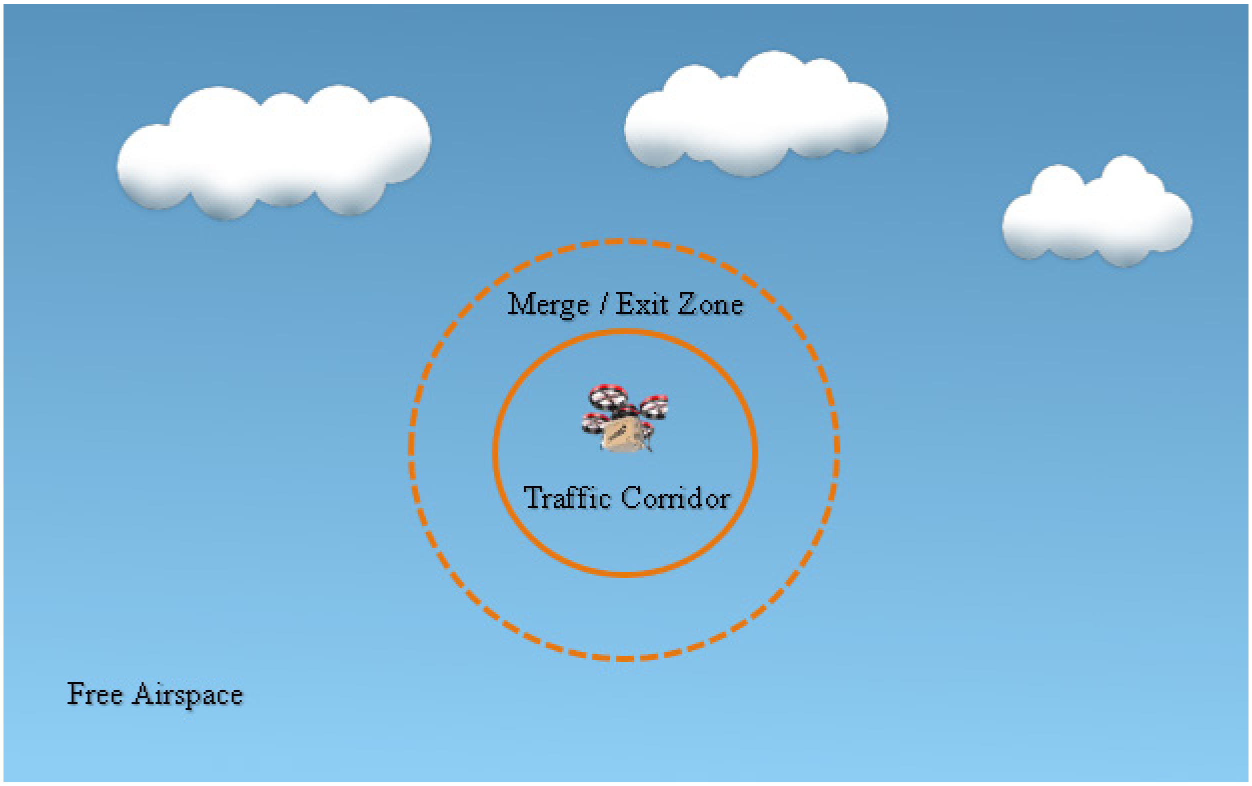


Figure 8: Cross-sectional Area of Corridor

Corridors would function similar to highways, and traffic volumes and densities could be calculated if necessary. However, if extra capacity is needed, the geo-fenced area could be expanded as necessary. Corridors could be viewed as arterial roads or freeways in the sky, while the free airspace surrounding them could be considered the local streets. Also as with highways, corridors should have buffer space between the opposing directions of travel. Of course, delivery drones need to return to the delivery hub as efficiently as they arrived at their current destination, so the returning route should be much the same as the original. One possible solution is to have a sizeable space between forward and reverse corridors where no traffic is allowed under any circumstances. *Figure 9* shows what this might look like.

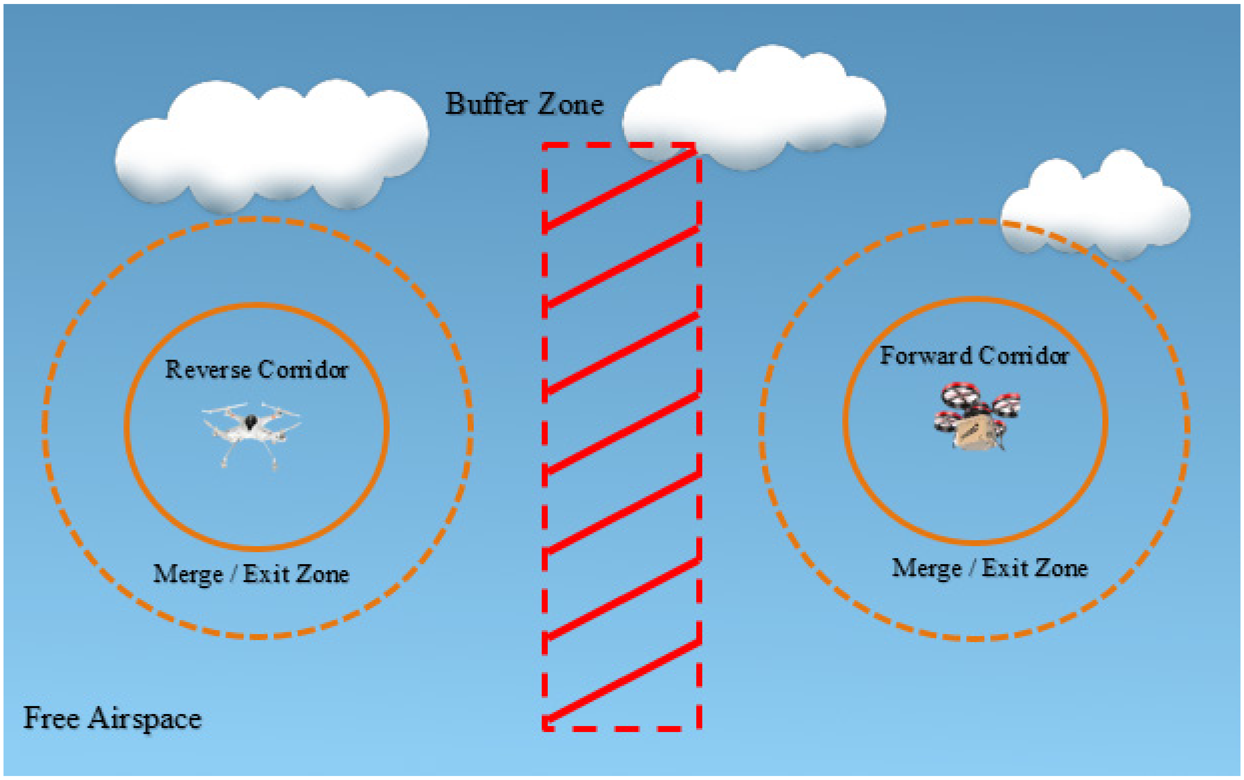


Figure 9: Cross-Sectional Area of Bi-Directional Corridor

The buffer zone between each direction of a corridor would serve to alleviate concerns of head-on collisions. The boundaries of the buffer zone would be geo-fenced to ensure that sUAS do not deviate from the small portions of free airspace between merge/exit zones and buffer zones and accidentally enter the buffer zones. Yet another issue that arises is the ability to pass within corridors. While establishing single lane corridors with set speed requirements (exact speeds to travel at, rather than speed limits) in each direction would be the obvious solution, this also presents more problems. Since there are expected to be a wide variety of sUAS models using the corridors, there will be many variations in top speed and overall maneuverability. From a policy perspective, only vehicles which meet certain requirements could be allowed to use corridors. However, companies which pay additional money for vehicles with higher top speeds should be given the ability to pass. Furthermore, emergency vehicles could be able to use a "passing lane" to reach a destination quickly without much disturbance to other vehicles in the airspace (this could prevent an "all-land" scenario to give way to emergency vehicles). Other vehicles in the corridor would simply shift to the "right lane" to allow emergency vehicles to pass. This setup has been proven to work well for vehicular traffic on four lane highways (two lanes in each direction).

For sUAS corridors with multiple lanes, it makes sense to stack the lanes vertically, rather than horizontally, since vehicles are not limited to a 2-D plane. In order to reach a passing lane, sUAS would enter an acceleration and ascend zone (or a deceleration and descend zone for returning to the normal traffic lane) and accelerate in order to match the speed of traffic. The passing lanes and ascend/descend zones would be surrounded by no-fly zones so that sUAS can only reach them from the standard lane ("right lane") of the corridor. Therefore, sUAS would not be able to enter the passing lane from free airspace, much like cars typically do not enter interstate highways into the left lane. *Figure 10* shows what corridors with several lanes could look like.

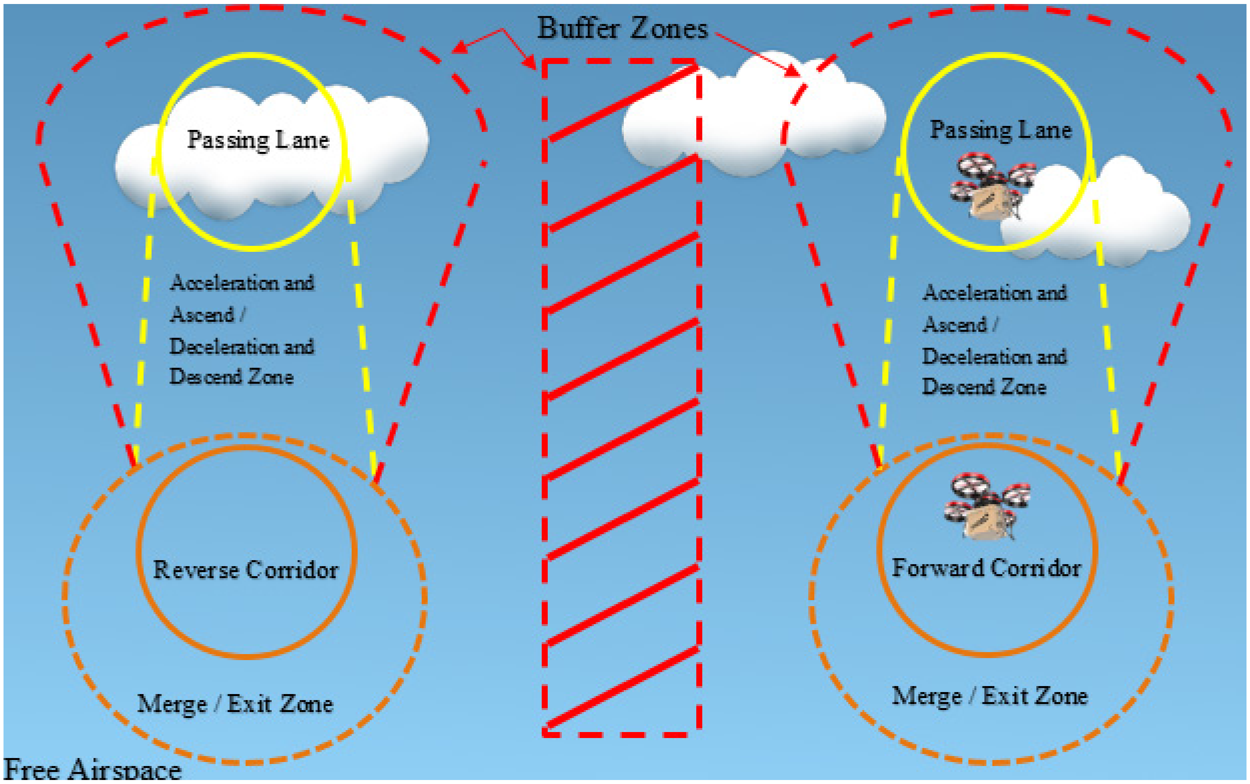


Figure 10: Multi-lane Corridors

We should note that corridors do not need to require defined speeds of all sUAS. Since all sUAS are expected to have their own geo-fences, following vehicles would autonomously slow down to the speed of the leading vehicle or enter the passing lane. This is very similar to adaptive cruise control in cars, which, when enabled by the driver, sets the vehicle at a constant speed and slows to the speed of the leading vehicle if sensors detect that their own vehicle is approaching the leading one too quickly. As a result, each lane of a corridor could either have a defined speed that all vehicles must travel at, or a bare minimum speed. In the defined speed scenario, the normal lane would require all vehicles to travel at 40 mph, and the passing or fast lane would require all vehicles to travel at 60 mph, for example. For the bare minimum speed scenario, the normal lane would require vehicles to travel at least 40 mph. When a vehicle traveling at 50 mph approaches a vehicle traveling at 40 mph from the rear, it could ascend into the passing lane and descend back to the normal lane once a safe clearance is established. Passing lanes could also have a minimum speed limit and a maximum speed limit.

Lastly, since free airspace is able to be reserved by those operating sUAS for purposes other than reaching a destination as quickly as possible, certain zones must not be able to be reserved as to limit the effectiveness of so-called destination sUAS. For example, free airspace surrounding corridors must always be open to all destination sUAS. Residential airspace, above private residences, is off-limits as to ensure the privacy and safety of residents. It is unclear how package deliveries will be permitted, but it is expected that individuals who sign up for quick shipping services which require sUAS will sign an electronic waiver to grant airspace rights to delivery drones directly above their properties, as mentioned before.

3.6 Proximity sUAS and Area of Operation

A trending usage of sUAS currently is for real estate agents to collect aerial pictures of properties on the market. With the aforementioned plan in place, the process for taking such photographs would be slightly different as to uphold safety and privacy. Say a real estate agent is attempting to sell a large two acre property in the suburbs. Assume that the current owner of the property has waived his or her right to the airspace for the real estate agent to operate freely on the property (this paper will not address the legal issues which stem from this). If the agent wants to obtain aerial photographs of the property for posting on the Internet, she can use her sUAS to photograph the property from above without any flight authorization so long as the elevation of the sUAS does not exceed 83 feet, and so long as she does not disobey local ordinances which dictate how close a private sUAS can approach an adjacent property and its private airspace. She will most likely have already acquired a license to operate sUAS.

Upon operating within the appropriate thresholds, the real estate agent discovers that she cannot capture photographs of the entire property without exceeding 83 feet. She then has three options to operate outside of the property's private airspace. She can seek permission from adjacent property owners to operate in their private airspace for a brief period of time to get a wider view of the property. The adjacent owners would notify the sUAS airspace manager (or some other overseer of sUAS operations) of their intentions to waive their private airspace to another operator, the real estate agent, if they were so inclined. The airspace manager would then inform the operator of the new boundaries to her flight and a time window for when she can operate. Another option would be to contact the sUAS airspace manager to reserve the free airspace directly above the 83 foot private airspace boundary. She could estimate that her sUAS would need to hover at 120 feet to obtain the photographs she needs. The airspace manager could then geo-fence a new volume of airspace above the property which extends to 140 feet to prevent other operators from interfering. Note that the real estate agent could operate in free airspace without making a reservation, but her sUAS would be required to have sense-and-avoid technology, and she would run the risk of coming into contact with package delivery and other sUAS in free airspace.

The third option would be to reserve extended free airspace. If the local ordinances proclaim that free airspace begins at 40 feet above public property, she could reserve the airspace (through the airspace manager) above the street from 40 to 100 feet, for example, or operate there without making a reservation if her sUAS is equipped with sense-and-avoid. Regardless of which option she chooses, she must launch her sUAS from the property she has permission to operate in, unless she has special permission from the municipality to launch from public land, such as the street, sidewalk, or a parking lot. Any of the three options for leaving private airspace might require one or more days of preparation. However, if she continues to operate her sUAS within the private airspace of the property she has been approved to sell, then she can launch and operate her sUAS as she pleases.

Conclusion

The task of organizing the low-altitude physical airspace for sUAS is somewhat complicated. In addition to preventing sUAS from crashing with one another, the privacy and safety of those on the ground must also be an important consideration. While Amazon and NASA have made early progress in this subject area, there was still a large gap in the combined vision of the two entities. The first step in establishing a proposal for the sUAS physical airspace design required the separation of sUAS into two distinct categories: destination sUAS and proximity sUAS. In order to design the airspace, one needs to know what to design for. Destination sUAS are those which seek to reach one or more destinations and return to base as quickly as possible. This allowed for the formulation of corridors, which permit destination sUAS, such as package delivery vehicles, to travel quickly and safely in file through the upper reaches of free airspace. Proximity sUAS are those which do not have a definite flight path but require a certain volume of airspace to achieve their goals, such as photography sUAS. Reserved airspace and private airspace are meant for proximity sUAS to operate safely without interference from other sUAS.

Private airspace is expected to extend to 83 feet above the property elevation, unless the structure on the property requires adjusting the top-end boundaries. The lateral boundaries will not be virtual lines immediately above the property lines, but rather a smaller outline of the property, shrunken by a distance determined by local ordinances as to protect the privacy of adjacent property owners. Property owners can waive their exclusive airspace rights to allow for package deliveries or other sUAS services. Above the 83 foot mark is free airspace, in which all sUAS must have sense-and-avoid technology and operate at a relatively low speed limit. Proximity sUAS users may reserve free airspace for some purpose so long as they do not prevent other operators from performing their duties. Destination sUAS users may reserve free airspace for extended periods of time in areas they frequently travel through. The expectation is that some airspace manager will oversee airspace reservations. Above public land, local ordinances will determine how low free airspace extends.

Finally, corridors for destination sUAS will be established in high traffic airspace to safely funnel vehicles between two well-traveled points. Destination sUAS will use the outer radius of the corridor to merge and reach the set speed required by traffic in the corridor. Once in the corridor, the sUAS must follow the speed limit, if there is one, or cruise at a safe distance from the preceding sUAS. Some corridors might have passing lanes for faster sUAS, complete with acceleration/deceleration zones to safely reach and return from them. The passing lanes will only be accessible from the main corridor, much like how highways do not let traffic enter in the left lane. When a Destination sUAS wants to leave a corridor and reenter free airspace, it must use the outer radius of the corridor to exit and slow to the speed required by vehicles in free airspace.

REFERENCES

CBSNews. (2015, Aug. 10). *Drones Interfering with Emergency Wildfire Responders.* Retrieved from http://www.cbsnews.com/news/drones-interfering-with-emergency-wildfire-responders/

De Los Santos, V., & Rios, J. (2015, June 11). *NASA UTM*. Retrieved from http://utm.arc.nasa.gov/index.shtml

Dillingham, G. L. (2015, May 11). *Unmanned Aerial Systems Status of Test Sites and International Developments. Testimony Before the Subcommittee on Aviation, Operations, Safety, and Security, Committee on Commerce, Science, and Transportation, U.S. Senate.* Retrieved from http://www.gao.gov/assets/670/669214.pdf

Golson, J. (2015, Aug. 4). *Do We Really Want Amazon's Drones to Swarm Our Skies?* Retrieved from http://www.wired.com/2015/08/really-want-amazons-drones-swarm-skies/

Henn, S. (2014, May 30). *Drone Wars: Who Owns The Air?* Retrieved from http://www.npr.org/sections/money/2014/05/30/317074394/drone-wars-who-owns-the-air

Hottman, S. B., Hanson, K. R., & Berry, M. (2015). *Literature Review on Detect, Sense, and Avoid Technology for Unmanned Aircraft Systems.* Federal Aviation Administration.

Munoz, C. A., Narkawicz, A., Chamberlain, J., Consiglio, M., & Upchurch, J. (2015, Sept. 13). *A Family of Well-Clear Boundary Models for the Integration of UAS in the NAS.* Retrieved from http://www.researchgate.net/publication/269193899_A_Family_of_Well-Clear_Boundary_Models_for_the_Integration_of_UAS_in_the_NAS

NASA. (2015). *Proc. of NASA UTM 2015: The Next Era of Aviation*. NASA Ames Research Center, Moffett Field, CA.

JOURNAL OF UNMANNED AERIAL SYSTEMS

PEER-REVIEWED ARTICLE

ASSESSMENT OF STRUCTURE FROM MOTION (SFM) PROCESSING PARAMETERS ON PROCESSING TIME, SPATIAL ACCURACY, AND GEOMETRIC QUALITY OF UNMANNED AERIAL SYSTEM DERIVED MAPPING PRODUCTS

Mr. Adam Benjamin
Geomatics Specialist, University of Florida Geomatics Program, Fort Lauderdale R.E.C., Fort Lauderdale, FL

Mr. Dennis O'Brien
MTA-Long Island Rail Road, Queens, NY

Dr. Grenville Barnes
Professor, University of Florida Geomatics Program, Gainesville, FL

Dr. Benjamin Wilkinson
Assistant Professor, University of Florida Geomatics Program, Gainesville, FL

Mr. Walter Volkmann
President, Micro Aerial Projects L.L.C., Gainesville, FL

ABSTRACT

As unmanned aerial system (UAS)-related mapping applications grow in number, a corresponding demand for enhancing field to finish UAS mapping workflows is at the forefront of the geospatial industry. This study investigates the impact processing parameter selection used in UAS multi-view stereo (MVS) photogrammetry processing has on the spatial accuracy and geometric quality of UAS-derived orthophotos and digital elevation models. The goal is to temporally optimize the semi-automated workflow by applying an understanding of the tradeoffs between parameter values and accuracy/quality metrics associated with the derived geospatial datasets. With 48 trials using the UAS-MVS representative software package, PhotoScan, results show that less rigorous structure from motion (SfM) processing parameters, specifically alignment and dense cloud generation parameters, can provide time savings without sacrificing the spatial accuracy of UAS-derived mapping products in low to moderate topographic relief areas. Lower 'quality' settings in the dense cloud generation phase led to the most significant time savings. When considering geometric quality in addition to spatial accuracy, reducing the alignment 'accuracy' and the number of key points does not impact the spatial accuracy of the resultant geospatial datasets.

I. Introduction

Through a combination of hardware and software designed to automatically collect geospatial data from an aerial platform over a user specified area, unmanned aerial systems (UAS) are a rapidly evolving technology with applications across numerous disciplines (Colomina & Molina, 2014; Pajares, 2015). New investigations of UAS technology for aerial mapping that were previously cost-prohibitive at smaller project scales emerged from development of cost-effective mapping sensors (e.g., consumer grade off-the-shelf (COTS) digital cameras) and user friendly image processing software utilizing structure-from-motion (SfM) and photogrammetric techniques (Sauerbier, Siegrist, Eisenbeiss, & Demir, 2011). Multi-view 3D reconstruction or multi-view stereo (MVS) photogrammetry algorithms provide the foundation for these hybrid SfM/photogrammetry software packages (Harwin & Lucieer, 2012). UAS-MVS derived mapping products are capable of filling a niche in the rapid collection of geospatial datasets at smaller project scales with greater temporal resolution and higher spatial resolutions than conventional aerial surveys or satellite imagery analyses (Casella et al., 2016; Messinger, Asner, & Silman, 2016). Further, Gerke and Przybilla (2016) summarized the main advantages of mapping with UAS over manned aircraft using three concepts: 1) flexibility through highly individualized flight patterns, 2) higher image resolution, and 3) ease of use due to minimal training requirements.

For geospatial professionals working on these smaller project scales, the typical platform is a small UAS (sUAS) (Colomina & Molina, 2014). The sUAS definition varies depending upon the aviation authority. For example, sUAS weigh less than 55 lbs. (24.95 kg) per the Federal Aviation Administration (FAA) in the United States (FAA, 2016) and weigh less than 20kg without fuel per the Civil Aviation Authority (CAA) in the United Kingdom (CAA, 2016). For most commercial mapping applications, sUAS weigh significantly less than these upper weight thresholds. The term “system” in “UAS” refers to the ground-based hardware and crew, airborne platform, and sensor suite used in the mapping process. Two primary sUAS sensor suite categories exist: remote sensing and positioning. The remote sensing category includes both passive (e.g., the aforementioned COTS digital cameras) and active (e.g., LiDAR) sensors that capture imagery and spatial data for the resultant mapping products. Positioning sensors provide navigation for the UAS and estimates of sensor locations can be used for processing the imagery. The small size of the UAS, correspondingly smaller format size of UAS imaging sensors, and lower altitudes result in smaller ground area captured per image relative to conventional aerial imagery from manned aircraft or satellites. Further, UAS-MVS algorithms require more image overlap (Turner, Lucieer, & Wallace, 2014) in order to properly reconstruct the imaged scene in three dimensions than classical photogrammetric methods (Wolf & Dewitt, 2000). Thus, UAS field operations acquire larger quantities of high resolution images for image processing. Depending upon the project area, the imaging sensor used, the mission altitude, and the processing parameters, UAS-MVS image processing can take over a day with high-end computers (Tavani et al., 2014). These long processing times can significantly reduce efficiency of UAS mapping operations.

Using high resolution UAS imagery to map small project areas successfully is a confluence of hardware enhancements (e.g., miniaturization for UAS field operations) and software advancements (e.g., robust algorithms that incorporate computer vision and photogrammetric principles). Low cost sensors capturing imagery for high resolution geospatial datasets on an as-needed basis stimulate growth in the variety and number of UAS mapping applications. Tavani et al. (2014) noted that these factors enabled a larger proportion of geoscientists to use these SfM-generated geospatial datasets because the barriers to utilization (e.g., cost, expertise) are much lower than existing technologies such as LiDAR and satellite imagery. For example, Fonstad et al. (2013) showed that SfM generated datasets have a spatial accuracy that compares favorably to aerial LiDAR datasets at a fraction of the sensor and acquisition costs for geomorphological applications. In forestry applications, Wallace et al. (Wallace, Lucieer, Malenovský, Turner, & Vopěnka, 2016) also showed that SfM generated datasets could be used to adequately characterize vertical vegetation structure relative to LiDAR at a much lower price point. When compared to satellite imagery acquisition, (Paneque-Gálvez, McCall, Napoletano, Wich, & Koh, 2014) summarized the cost benefits of using drones for on demand imagery acquisition in forested communities. Further, Colomina and Molina (2014) address the widespread adoption of UAS technology by summarizing the increase in volume of annual UAS-related peer reviewed publications from 2005 to 2013. Even as adoption of UAS-MVS technology increases, significant hurdles still exist. One of the most critical obstacles to enhanced efficiency in the UAS mapping workflow is the computer resource intensive nature of UAS-MVS software (Barnes & Volkmann, 2015; Dietrich, 2016; Turner et al., 2014). Thus, UAS image processing time efficiency is a research area in need of further exploration.

For UAS-MVS image processing, users choose from a multitude of hardware/software options. Some users access high powered servers in a cloud processing environment while others rely on consumer grade laptop specifications. The hardware/software choices greatly influence UAS-MVS processing time. Turner et al. (2014) investigated three potential SfM processing solutions: 1) a commercial software package on a high end desktop computer, 2) an open source software package in a local server environment, and 3) a commercial cloud processing solution. The authors found that processing efficiency depended on using multiple central processing units (CPUs) (i.e., commercial software had multi-CPU ability while open source software did not have that functionality). Additionally, internet upload speeds greatly impacted overall processing time from field to finish for the cloud based processing solution (Turner et al., 2014). In some applications such as cadastral mapping in developing countries (Barnes & Volkmann, 2015), cloud based processing solutions are not feasible due to the large size of data sets and the lack of reliable internet connections. Thus, the optimization of UAS-MVS processing parameters for local computer processing is necessary to generate quality geospatial datasets in a time efficient manner.

A systematic approach to UAS-MVS processing parameter selection is still evolving in the literature. Previous processing parameter approaches with a common SfM processing solution, Agisoft Photoscan (Agisoft, 2016), tended to either 1) briefly mention a trial and error method (Gross & Heumann, 2016; Puliti, Olerka, Gobakken, & Næsset, 2015; Turner et al., 2014), 2) select parameters for optimal accuracy regardless of processing time (Messinger et al., 2016), or 3) base parameter selection on user manual recommendations (Casella et al., 2016; Dandois & Ellis, 2013; Puliti et al., 2015; T. N. Tonkin, Midgley, Graham, & Labadz, 2014). More recent studies have focused on improving UAS data acquisition techniques (e.g., flight planning (Carbonneau & Dietrich, 2016; Fonstad et al., 2013; Mike R. James & Robson, 2014), configuring ground control networks (Clapuyt, Vanacker, & Van Oost, 2016; M. R. James, Robson, d'Oleire-Oltmanns, & Niethammer, 2017; Toby N. Tonkin & Midgley, 2016) and refining camera calibration methods for non-metric cameras (M. R. James et al., 2017; Mike R. James & Robson, 2014). The adoption of these methods going forward will lead to more accurate and complete SfM-derived geospatial datasets. However, a study emphasizing a systematic approach to parameter selection for enhancing processing time efficiency without sacrificing accuracy or quality is still lacking.

The primary objective of this paper is to investigate the impact that SfM/UAS-MVS processing parameters have on spatial accuracy and geometric quality of UAS-derived geospatial datasets. Spatial accuracy, described in greater detail in Section 2.6, assesses the horizontal and vertical difference between ground truth data and the final mapping products. Meanwhile, geometric quality, described in greater detail in Section 2.7, assesses the qualitative fidelity of the derived orthophoto mosaics in terms of distinguishing and identifying features within the imagery. The goal is to temporally optimize the semi-automated workflow by applying an understanding of the tradeoffs between parameter values and accuracy/quality metrics associated with the UAS-derived geospatial datasets.

2. Materials and Methods

2.1. UAS Mapping Workflow

To efficiently obtain quality geospatial datasets from UAS mapping platforms, a workflow similar to Figure 1 needs to be established and followed. The UAS mapping workflow has three phases: 1) mission planning/field operations, 2) aerotriangulation/processing, and 3) geospatial dataset creation. The first phase involves the necessary UAS mapping data collection techniques for both the establishment of ground control and the airborne operations. The second phase covers the integration of photogrammetric principles such as aerotriangulation (i.e., determination of XYZ coordinates of individual points based on photo coordinate measurements (Wolf & Dewitt, 2000)) with SfM processes. Lastly, the generation of the UAS-derived geospatial datasets occurs in the third phase.

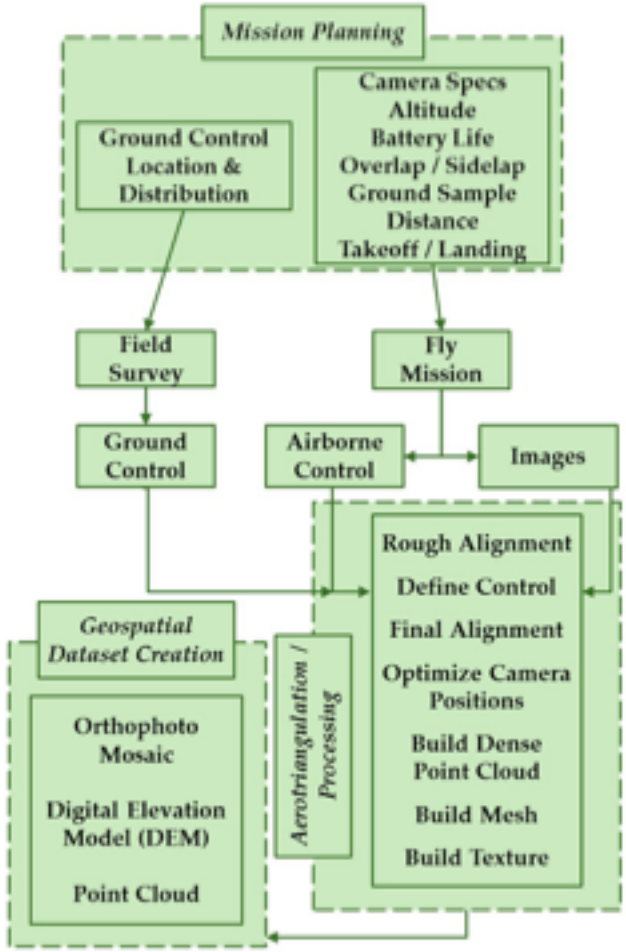


Figure 1: Unmanned aerial systems mapping workflow

2.2. Study Site

The mission planning and field operations based components of the UAS mapping workflow are location dependent. Physical site conditions, airspace regulations, weather conditions, and equipment capabilities factor into operational considerations. The 3.5 ha study area, outlined in blue in Figure 2, was located in Alachua County, FL, USA. The study area contained a high point density of field surveyed three-dimensional and vertical-only ground control relative to areas outside of the study area boundaries. Thus, the authors selected the study area for two reasons: 1) the availability of high density ground control and 2) the reduction of data storage (e.g., 122 images for the study area versus 314 images for the entire site) needed to conduct the numerous trials in this study. While study area selection affected absolute processing time, it did not affect relative processing times amongst the trials. The relative impact each processing parameter had on processing time within and across trials was preserved. Section 2.5 discusses the processing parameters and trials in greater depth.

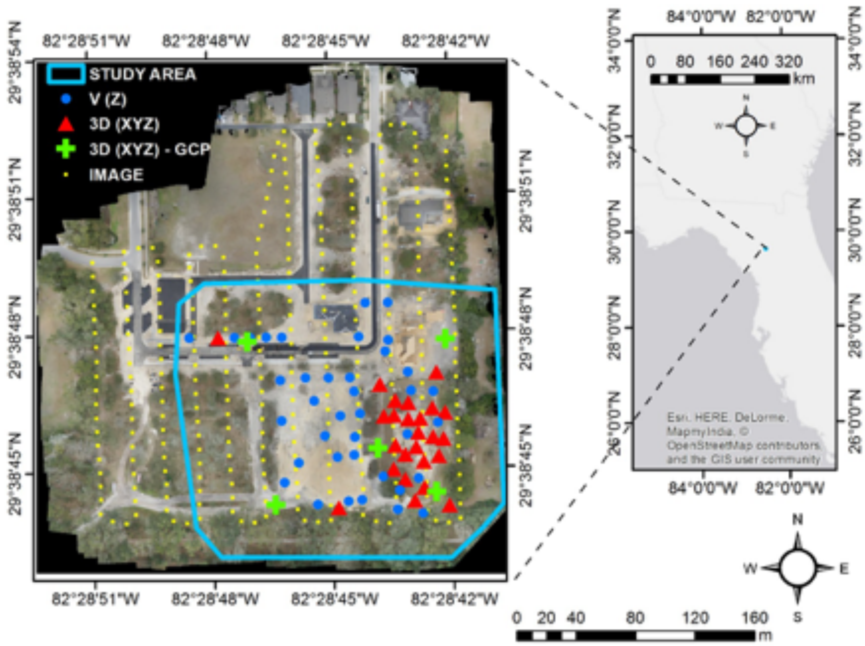


Figure 2: The study area (blue polygon) in Alachua County, FL, USA has vertical ground control (blue circles) and 3D ground control (red triangles) for spatial accuracy evaluation, 3D ground control (green crosses) for georeferencing the imagery, and approximate image locations (yellow squares) for showing the flight acquisition pattern.

2.3. Field Data Collection

One UAS flight with a MAP-M4 multirotor quadcopter airframe as shown in Figure 3 was flown over the entire site on February 15, 2016. The sUAS had a single-frequency code based GNSS positioning sensor that navigated the aircraft over a predetermined flight plan. The yellow squares in Figure 2 show the approximate locations where nadir images were acquired along the north-south flightline. The imaging sensor was a 16mm focal length Sony a6000 camera with an effective image resolution of 6000 × 4000 pixels. Given an altitude above ground level (AGL) of about 55m, the ground resolution of the resultant sUAS imagery was ~1.25 cm/pixel.

Figure 3: MAP-M4 multirotor quadcopter UAS

Five field surveyed aerial targets were used as ground control points for georeferencing the imagery and resultant geospatial datasets. These targets, as shown in Figure 4, consisted of four alternating black and white triangles intersecting at the center of the 25 cm square targets. Field survey observations also included measurements to additional horizontal aerial target check points (e.g., 10.5 cm white circular disks) and vertical only check points. The aerial targets provided photo identifiable features used as three-dimensional (i.e., both horizontal and vertical) check points when overhead vegetation did not obscure the targets. Lacking photo-identifiable features, the high accuracy vertical only check points comprised the primary reference data in the digital elevation model (DEM) spatial accuracy evaluation. Surveying with both V-Map dual frequency GNSS receivers and a Spectra Precision Focus 35 robotic total station as shown in Figure 4 enabled establishment of the entire ground control point (GCP) network. Two Alachua County corner markers with published Florida North State Plane Coordinates provided the geodetic control onsite. Static GNSS survey observations and subsequent GNSS post processing with RTKLIB software (RTKLIB, 2017) verified the observed baseline distance matched the computed baseline distance from the published coordinates. Using this geodetic control, redundant robotic total station measurements yielded a horizontal and vertical precision of less than 0.01 m for control and check point data.

(a) 

(b) 

Figure 4: Black and white square aerial targets (a) were field surveyed using a robotic total station (b) and GNSS surveying.

2.4. Computer Resources

The “aerotriangulation/processing” and “geospatial dataset creation” phases of the UAS mapping workflow in Figure 1 require substantial computer resources to process high resolution sUAS mapping products (Verhoeven, 2011). A 64-bit Windows 7 desktop computer with a 3.70 GHz Intel Xeon CPU E5-1620 processor, 48GB of RAM, a 1TB 7200 RPM hard drive, and a NVIDIA Quadro K4000 graphics processing unit (GPU) performed all data processing during the study. These specifications meet or exceed the Agisoft recommended configuration of a 64-bit Windows OS, Intel i7 processor, 12GB RAM, and a NVIDIA GeForce 8xxx series graphics card (Agisoft, 2016).

There are several software packages capable of processing SfM-MVS datasets including Agisoft Photoscan (Agisoft, 2016), Pix4D (Pix4D, 2016), and VisualSFM (Wu, 2016). The authors chose Photoscan (v1.2.5 build 2614) as the representative software package for this study because: 1) it is widely used within both academia and industry for generating UAS-derived geospatial datasets (Carbonneau & Dietrich, 2016; Casella et al., 2016; M. R. James et al., 2017; Javernick, Brasington, & Caruso, 2014; Puliti et al., 2015; Tavani et al., 2014; Toby N. Tonkin & Midgley, 2016; Turner et al., 2014; Verhoeven, 2011), 2) it is an attractive option for new UAS mapping technology adopters due to the extensive software user community (http://www.agisoft.com/forum/) and manageable purchase price for small organizations (Gross & Heumann, 2016), and 3) it has extensive reporting features related to processing time (Agisoft, 2016). Following the Agisoft Photoscan User Manual’s recommendations for optimal performance, one central processing unit (CPU) core was deactivated for each GPU in use (Agisoft, 2016). Due to Agisoft support of OpenCL acceleration during the depth maps reconstruction phase of building dense point clouds (Agisoft, 2016), processing speed can increase substantially using the multiple GPU/CPU approach (Turner et al., 2014; Verhoeven, 2011). Thus, the optimal processing unit configuration for this system was seven CPUs and one GPU.

2.5. SfM Processing

In the aerotriangulation/processing phase of the sUAS mapping workflow shown in Figure 1, the seven distinct steps in the Agisoft Photoscan workflow are: (1) rough alignment, (2) control definition, (3) final alignment, (4) camera position optimization, (5) dense point cloud generation, and optionally, (6) mesh generation, and (7) texture generation. These processes (6 and 7) enable the generation of digital elevation/surface models and the orthophoto mosaics for traditional geospatial data products; however, these products may not be required for every research application.

The following is a brief description of the SfM processing workflow within Photoscan. Additional details can be obtained from the Agisoft user manual (Agisoft, 2016) and previous literature on SfM processing (Casella et al., 2016; Puliti et al., 2015; Verhoeven, 2011). Upon importing all images into Photoscan, the rough alignment process benefited from the manual control definition of three unique GCPs. Manual control definition entailed selecting the center of the photo-identifiable field surveyed ground control where the GCP was visible. Each GCP was selected in two separate images. Rough alignment stitched the images together into a block using tie points, which are described in more detail in Section 2.5.1.

While control definition in an automatic or semi-automatic manner is possible depending upon the photo-identifiable aerial targets and the centroid defining algorithms (Harwin & Lucieer, 2012), the only manually completed step in the sUAS workflow within Photoscan v1.2.5 was control definition. While Photoscan now supports coded targets for automatic aerial target identification, this study did not use this option. The rough alignment step expedited additional manual control definition by enabling the software to display approximate locations of all targets in the images based on the location of the 3 GCPs used to georeference the image block in the rough alignment step.

For all trials, five three-dimensional GCPs provided the georeferencing for the final geospatial datasets. As shown by the green crosses in Figure 2, the GCP spatial configuration approximates a bounding square with a point in the middle. Based on previous internal research, this spatial configuration provided the optimal configuration for a five point ground control network with diminishing returns on spatial accuracy for additional control points used in the georeferencing. Recent literature discusses optimal GCP distribution in greater depth and provides support for this GCP configuration through emphasis on strong network geometry with a spatial distribution spanning the survey area (M. R. James et al., 2017). Specifically, Tonkin and Midgley (2016) found that 4 or more GCPs well distributed throughout the study area provided vertical spatial accuracies consistent with the positional quality of the measurements used to determine the GCP coordinates.

For processes (3-7), an investigation into reducing Photoscan processing parameter rigor from the default settings shown in Table 1 and the corresponding effects of these altered settings on spatial accuracy, geometric quality, and processing time ensued. Table 2 shows an example of the processing time for this study site using Photoscan default parameters (Agisoft, 2016). Of the remaining five steps in the processing phase, final photo alignment (step 3) and dense cloud generation (step 5) accounted for over 95% of total processing time. Therefore, thorough investigation from here on focused primarily on alteration of the corresponding parameters for those two steps.

Table 1: Default Agisoft Photoscan processing parameters

Alignment	
Accuracy	High
Pair preselection	Disabled
Key point limit	40000
Tie point limit	4000
Constrain features by mask	No
Camera Optimization	
Parameters	f, cx, cy, k1-k3, p1, p2
Dense Point Cloud Generation	
Quality	High
Filtering mode	Moderate
Mesh Generation	
Surface type	Height field
Source data	Dense Cloud
Interpolation	Enabled
Quality	High
Depth filtering	Moderate
Texture Generation	
Mapping mode	Adaptive orthophoto
Blending mode	Mosaic
Texture size	4,096 x 4,096
DEM Generation	
Source data	Dense cloud
Interpolation	Enabled
Orthomosaic Generation	
Channels	3, uint8
Blending mode	Mosaic
Surface	Mesh
Enable color correction	No

Table 2: Processing time for the study area using default Photoscan parameter settings

#	Parameter	Proc. Time (min)	Proc. Time (% Total)
1	Final Alignment	41.88	17%
2	Camera Position Optimization	0.03	0%
3	Dense Cloud Generation	191.25	79%
4	Mesh Generation	4.57	2%
5	Texture Generation	5.18	2%

2.5.1. Final Alignment & Dense Cloud Generation

Depending upon the project size and the parameters selected in Table 3, final photo alignment processing time can vary from a few minutes to days. Thus, a balance between alignment time and generated mapping dataset accuracy is necessary. Table 3 described the four primary parameters used in the alignment process: (1) 'accuracy', (2) 'pair preselection', (3) 'key point limit', and (4) 'tie point limit'. The 'accuracy' parameter does not denote the spatial accuracy of the alignment. Instead, 'accuracy' in the context of alignment for the Photoscan processing parameters defines the spatial resolution of each image used in the alignment process (e.g., a higher 'accuracy' value means an image with a higher spatial resolution is used). The 'pair preselection' parameter enables the use of precise camera exposure stations. Camera exposure positioning data was not available in the EXIF metadata for the images in this study. Enabling 'pair preselection' is recommended when the data is available as it improves processing speed in the alignment process. 'Key point limit' defines the maximum number of feature points to collect in each image while 'tie point limit' defines the maximum number of points matched across images.

Table 3: Alignment parameters in Agisoft Photoscan. The 'Evaluated' column indicates whether an alignment parameter was altered for evaluation during the processing trials.

Parameter	Value	Interpretation	Evaluated
Accuracy	Highest	Upscales image resolution by 4	Yes
	High	Original image resolution	Yes
	Medium	Downscales image resolution by 4	Yes
	Low	Downscales image resolution by 16	Yes
	Lowest	Downscales image resolution by 64	Yes
Pair preselection	Disabled	Ground control only	Yes
	Reference	Airborne control enabled	No
Key point limit	XXXXX1	Upper threshold of feature points per image	Yes
Tie point limit	XXXX	Upper threshold of matching points per image	Yes

1 - XXXXX denotes a 5 digit integer number

From the summary of dense cloud generation processing parameters in Table 4, the primary parameter is dense point cloud 'quality'. Similar to the alignment process, 'quality' corresponds to image resolution. The 'depth filtering' parameter indicates which algorithm is used to mitigate artifacts based on knowledge of the scene geometry. The authors chose 'moderate depth filtering' based on past experience with similar sites and to provide a balance between the 'mild depth filtering' approach that preserved more details with a subsequent increase in outlier data and the 'aggressive depth filtering' approach with cleaner data and fewer preserved details.

Table 4: Dense cloud generation parameters in Agisoft Photoscan. The 'ultra high quality' setting was tested against the 'high quality' setting by keeping the remaining SfM processing parameters the same. Testing results yielded a more than six fold increase in processing time ('ultra high quality' total time > 20 hrs). Thus, the 'ultra high quality' value was not evaluated for consideration as an optimal parameter.

Parameter	Value	Interpretation	Evaluated
Quality	Ultra High	Original image resolution	No
	High	Downscales image resolution by 4	Yes
	Medium	Downscales image resolution by 16	Yes
	Low	Downscales image resolution by 64	Yes
	Lowest	Downscales image resolution by 256	No
Depth filtering	Aggressive	Filters points because meaningful small details are not in subject area	No
	Moderate	Balance between aggressive and mild approaches	Yes
	Mild	Filters fewest points to preserve small details in subject area	No

The relationship between the different alignment and dense cloud parameters is explored herein through 48 separate trials. To identify the parameters that provided the best tradeoff between spatial accuracy and processing time, the experimental setup included the following characteristics. Each trial used the same five GCP configuration described earlier and shown in Figure 2. Further, the camera position optimization, mesh generation, texture generation, DEM generation, and orthophoto mosaic generation parameters were the same for all trials. Generated orthophoto mosaics and DEMs had spatial resolutions of 1.2 cm/pixel and 2.4 cm/pixel, respectively. The fourth column in Table 3 and Table 4 indicate whether a particular parameter was modified during the trials. Different combinations of 'key point limit' and 'tie point limit' used in the trials are shown in Table 5. Combination 1 with a 40,000 'key point limit' (KP) and a 4,000 'tie point limit' (TP) is the default setting for KP and TP alignment settings within Photoscan (Agisoft, 2016). Using the default 40,000 KP setting was a frequent occurrence within the literature (Casella et al., 2016; Gross & Heumann, 2016; M. R. James et al., 2017; Messinger et al., 2016; Puliti et al., 2015; Verhoeven, 2011). Combination 7 and combination 8 were included after considering experience with photogrammetric ratios of feature points to matching points as well as DEM errors with small tie point limits discussed in the results section.

Table 5: ‘Key point limit’ and ‘tie point limit’ combinations used during trials

#	Key Point Limit	Tie Point Limit
1	40000	4000
2	40000	2000
3	40000	1000
4	20000	2000
5	20000	1000
6	20000	500
7	8000	5000
8	4000	2500

2.6. Spatial Accuracy Assessment

The 2015 American Society of Photogrammetry and Remote Sensing (ASPRS) Positional Accuracy Standards for Digital Geospatial Data provide spatial accuracy reporting metrics related to horizontal and vertical accuracy (ASPRS, 2015). The root mean squared error (RMSE) is the primary spatial accuracy metric that computes the difference between a known/true value and an observed value. Easting RMSE (RMSEX), northing RMSE (RMSEY), horizontal RMSE (RMSER), and horizontal accuracy at a 95% confidence level (HZ95) are horizontal absolute accuracy metrics derived from comparing SfM-generated geospatial datasets to field surveyed ground control. Likewise, vertical RMSE (RMSEZ) and vertical accuracy at a 95% confidence level (V95) are the vertical absolute accuracy metrics.

To compute the GCP observed values, the Editor toolbar in ArcGIS 10.4 for Desktop (ArcMap) software from Environmental Systems Research Institute (ESRI) (ESRI, 2016) enabled the manual selection of the center of photo identifiable targets on each orthophoto mosaic. After digitizing all photo identifiable GCPs for each trial, a Python script used the ArcPy package from ESRI and the Field Calculator tool to obtain the manually measured coordinates. All horizontal coordinates were in the Florida North state plane projection, NAD83 (2011) horizontal datum. A comparison between these observed GCP coordinates and the field-surveyed GCP coordinates determined horizontal coordinate differences. Likewise, a comparison between the high resolution digital elevation model (DEM) derived from the sUAS data and the field-surveyed GCP heights formed the basis of the vertical accuracy evaluation. All vertical coordinates were in the same vertical datum, NAVD88 (GEOID 12B). The spatial accuracy metrics used the horizontal and vertical coordinate differences to compute the RMSE metrics. If necessary, the RMSE data is available to communicate the spatial accuracy in other positional guidelines such as Accuracy Standards for Large Scale Maps (ASPRS, 1990), Vertical Accuracy Reporting for LiDAR Data (Flood, 2004), or Geospatial Positioning Accuracy Standards: National Standard for Spatial Data Accuracy (NSSDA) (FGDC, 1998) standards.

Summary statistics including mean, standard deviation, skew, and kurtosis were calculated for each of the spatial accuracy metrics: easting RMSE (RMSEX), northing RMSE (RMSEY), horizontal RMSE (RMSER), horizontal accuracy at a 95% confidence level (HZ95), vertical RMSE (RMSEZ), and vertical accuracy at a 95% confidence level (V95). Per the Positional Accuracy Standards for Digital Geospatial Data (ASPRS, 2015), skewness and kurtosis are recommended to show the normality of the error distribution and the frequency of extreme errors. In general, skewness assesses the symmetry of the error distribution . Kurtosis assesses how much of the variance is the result of extreme deviations relative to modest sized deviations .

2.7. Geometric Quality Assessment

While spatial accuracy is critical for quantifying the uncertainty in the measurement of features in sUAS-derived geospatial datasets, geometric quality (i.e., image quality) assesses the qualitative fidelity of the derived orthophoto mosaics in terms of distinguishing and identifying features within the imagery. Gross and Heumann (Gross & Heumann, 2016) used image artifacts (e.g., outliers) and image blur to assess image quality. Their assessment methodology was adopted herein, where 100 points were randomly placed throughout the orthophoto mosaic. Buffers of 2.5m were drawn around each point creating search area circles with a 5m diameter. Figure 5 shows the random distribution pattern of these geometric quality assessment circles. Circle locations spanned a variety of land cover classes including bare ground, trees, structures, and roads. These circles were examined manually within ArcMap software for the presence of image artifacts and image blur. While image artifacts are regions of misaligned pixels within the image that have distinct visible errors, image blur is simply an area of the orthophoto mosaic that has a blurred appearance due to either input image quality or image processing (Gross & Heumann, 2016). The results are reported in a binary format as either presence (1) or absence (0) for each circle. In addition, data voids and the dominant land cover class were also noted. Figure 6 is an example of the presence of both an image artifact and image blur for one of the 100 samples. The same sample areas were used for analysis across the trials. Finally, Figure 7 shows a variety of land cover classes illustrating the presence of image artifacts or image blur.

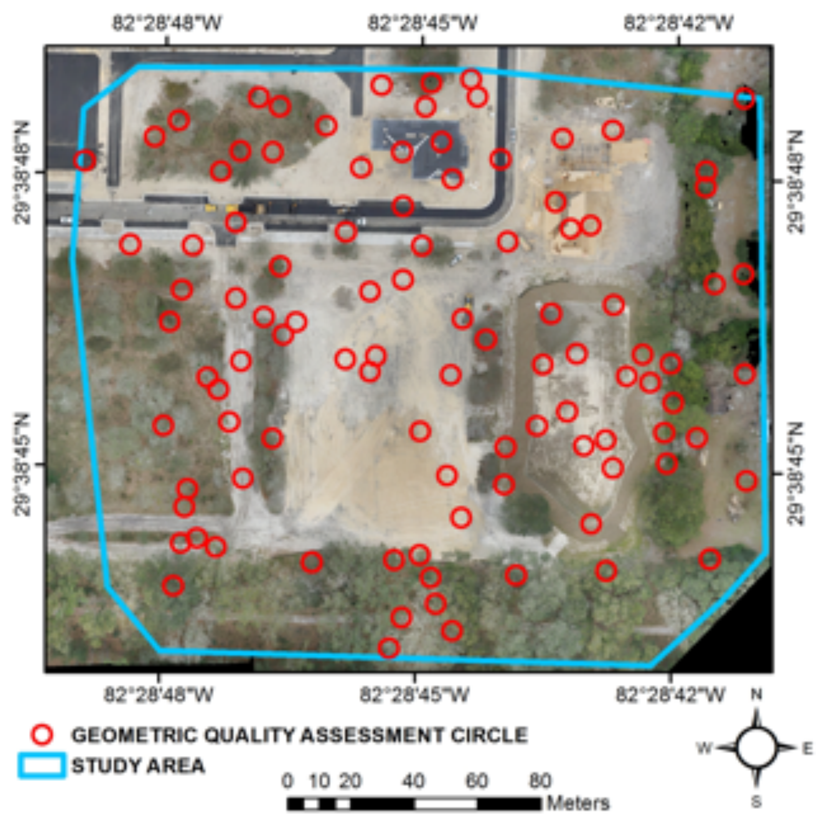


Figure 5: Random distribution of geometric quality assessment circle locations for identifying the presence of image artifacts, image blur, and data voids.

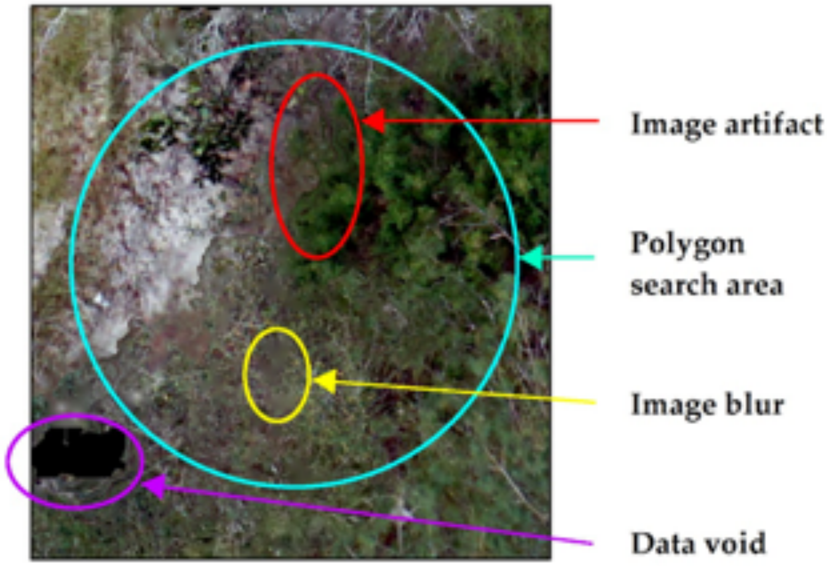


Figure 6: Example of geometric quality assessment 5m diameter circle (blue) showing an image artifact (red), image blur (orange), and a data void (purple). Note that the data void occurs outside of the assessment polygon.

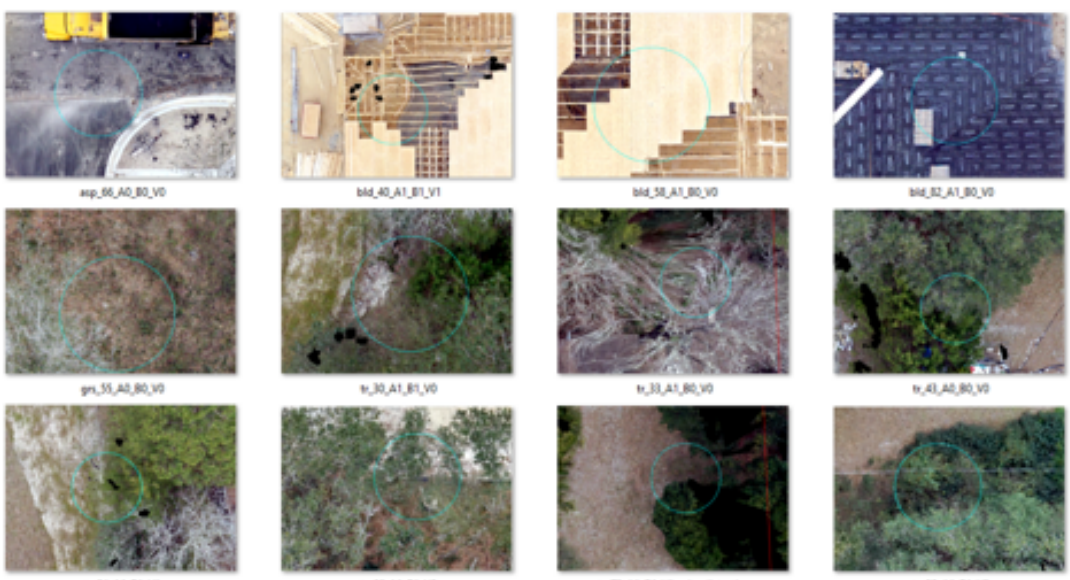


Figure 7: Examples of geometric quality assessment polygons organized by dominant land cover class: asphalt (asp), building (bld), grass (gr), and tree (tr). For the image labels, ‘A1’ indicates the presence of at least one image artifact within the circle, ‘B1’ indicates the presence of image blur within the circle, and ‘V1’ indicates a data void within the circle. Meanwhile, ‘A0’ indicates the absence of image artifacts within the circle.

3. Results

3.1. Spatial Accuracy Assessment Results

Summary statistics for the horizontal and vertical spatial accuracy assessment of the 48 trials are shown in Table 6 and Table 7, respectively. These results are displayed graphically in Figure 9(a-b). The mean horizontal accuracy assessment statistics in Table 6 (a-b) for each of the horizontal metrics show that the mean horizontal accuracy is about 1 cm with mm-level variation in the standard deviation of the mean across the 48 trials. Thus, horizontal accuracy is of a similar magnitude (~1 cm) to the error associated with the ground control survey described in Section 3.3. The low R-squared values in Figure 9a derived from the linear regression implemented in the Data Analysis ToolPak within Microsoft Excel supports the notion that horizontal spatial accuracy has a significant, weak negative correlation (coefficient = -0.0000049; p = 0.004) with processing time. Therefore, horizontal spatial accuracy is minimally affected by higher 'accuracy' or 'quality' Photoscan parameters causing the longer processing times. Thus, the lower 'accuracy' alignment parameters can be used with shorter duration processing times without sacrificing spatial accuracy of the geospatial datasets. Given minimal variation in horizontal spatial accuracy over all trials, the impact of altering the alignment 'accuracy', the 'key point limit', the 'tie point limit', and the dense cloud 'quality' has minimal impact on the resultant horizontal spatial accuracy. Further statistical analysis is necessary to parse out if the small variations that are present are due to one or two significant parameters. Lastly, the direction of flight was approximately north-south as shown in Figure 2. Therefore, it is interesting that the along flight line errors in the north-south direction are more normally distributed relative to the cross flight line easting errors which have a larger skew. Upon visual inspection of maps illustrating the spatial variability of horizontal spatial errors in each individual trial, no systematic distortions relating to the north-south flight pattern could be discerned. Figure 8 is an example of a spatial variability map for the default Photoscan settings from Table 1 with directional symbols showing errors scaled by 500. The random spatial pattern of errors found in this map was typical across the trials.

Table 6: Easting (a), northing (a), and horizontal (b) spatial accuracy assessment summary table for 48 trials

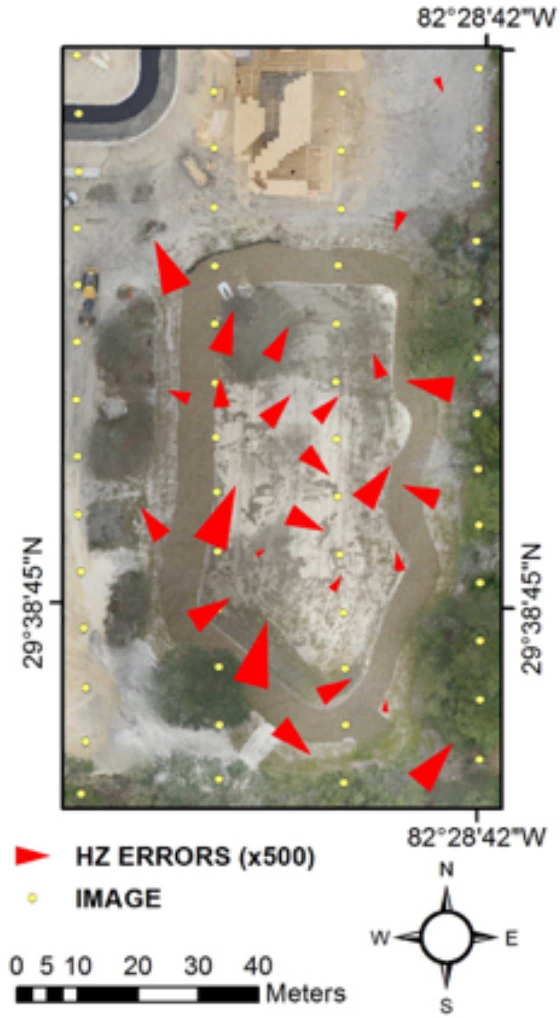
(a)

	Easting Accuracy - RMSEX			Northing Accuracy - RMSEY				
	avg (m)	std dev (m)	skew	kurtosis	avg (m)	std dev (m)	skew	kurtosis
Mean	0.008	0.006	1.50	3.63	0.006	0.004	0.47	-0.04
SD	0.001	0.001	0.74	3.46	0.001	0.001	0.49	1.14
Max	0.010	0.008	3.09	11.62	0.007	0.006	1.94	5.15
Min	0.005	0.004	0.25	-0.93	0.004	0.003	-0.47	-1.31
Range	0.005	0.003	2.84	12.54	0.003	0.003	2.41	6.46

(b)

	Horizontal Accuracy - RMSER				95% Conf. Level		GCPs	Proc. Time
	avg (m)	std dev (m)	skew	kurtosis	Hz (m)	Hz	(min)	
Mean	0.010	0.006	1.17	2.82	0.018	22.0	111.8	
SD	0.001	0.001	0.67	2.42	0.002	0.8	82.3	
Max	0.013	0.009	2.50	8.52	0.022	25	247	
Min	0.009	0.004	-0.25	-0.51	0.016	21	10	
Range	0.004	0.005	2.75	9.03	0.006	4	237	

Figure 8: Spatial variability error map for the default Photoscan settings with directional symbols showing horizontal spatial accuracy errors scaled by 500.



The vertical accuracy assessment statistics in Table 7a for the two vertical metrics show that there is more variation in vertical accuracy than horizontal accuracy. It is known in photogrammetry that vertical accuracy will perform worse than horizontal accuracy due to vertical measurements requiring multiple images (i.e., ray intersections) while horizontal measurements can be made in one image (Wolf & Dewitt, 2000). However, much of the vertical accuracy variation can be attributed to the three outlier trials shown in Figure 9b. Each of these trials used the 'highest accuracy' alignment, 20000 'key point limit', and 500 'tie point limit' with varying 'quality' dense clouds. Figure 10b shows the errors in the digital elevation model near a building roof. The likely error source is the use of a relatively small number of tie points with upscaled imagery. With an increased number of pixels per image, excessive matching in specific areas of the image could result in an unequal distribution of tie points and, in turn, errors in the alignment. This would affect subsequent processing steps including the digital elevation model generation. With the three trials removed from the testing as shown in Table 7b and Figure 9b, the vertical accuracy variation is more stable across the remaining 45 trials with 2 mm and 4 mm standard deviations for the mean vertical spatial accuracy metrics. With the removal of the 3 outlier trials, a simple linear regression became possible due to dataset normality. The low R-squared value in the vertical accuracy Figure 9b for the 45 trial subset and the corresponding high p-value (0.677) for processing time means processing time does not significantly impact vertical spatial accuracy. Thus, higher 'accuracy'/'quality' processing parameters and longer processing times do not lead to substantial improvements in vertical spatial accuracy.

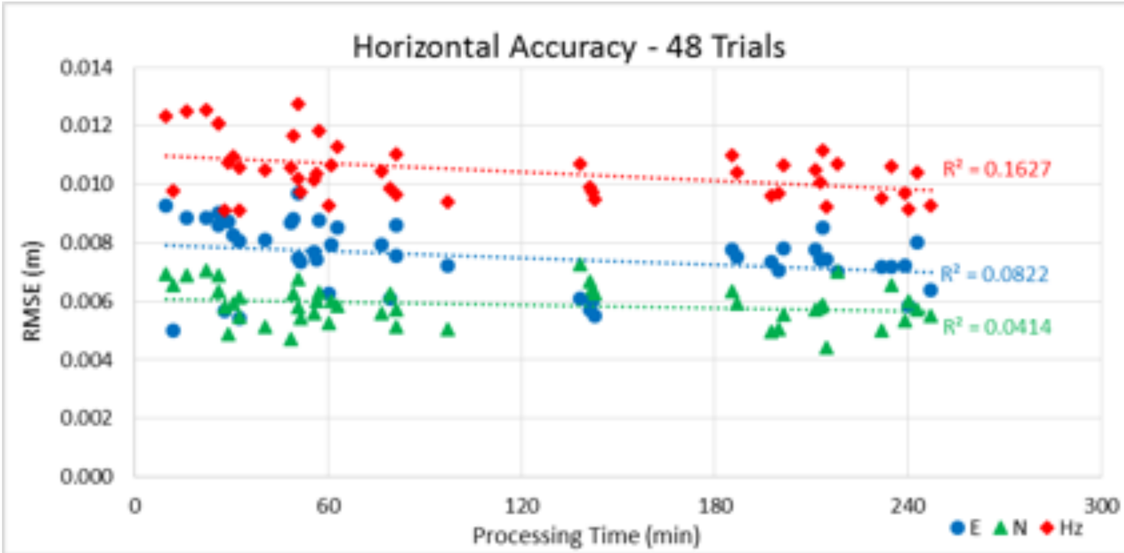
Table 7: Vertical spatial accuracy assessment summary table for trials. (a) All 48 trials. (b) The 45 remaining trials after removing the three outlying trials.

(a)

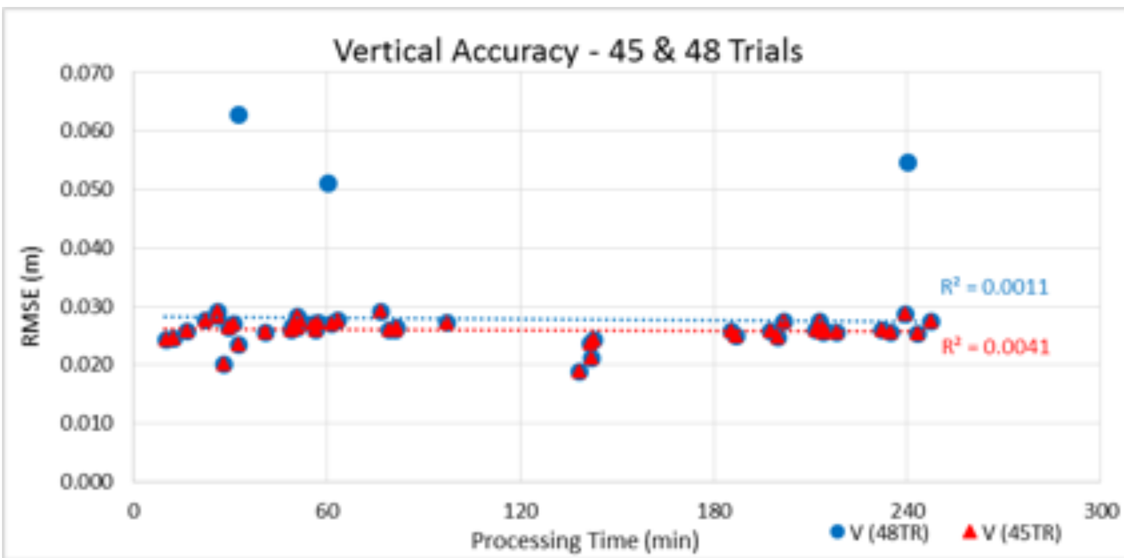
	Vertical Accuracy - RMSEZ			95% Conf. Level		GCPs	Proc. Time
	avg (m)	std dev (m)	skew	kurtosis	V (m)	V	(min)
Mean	0.028	0.031	1.470	5.673	0.055	62.9	111.8
SD	0.008	0.050	1.780	14.979	0.015	0.3	82.3
Max	0.063	0.263	7.769	61.129	0.123	63	247
Min	0.019	0.014	0.312	-0.761	0.037	62	10
Range	0.044	0.249	7.457	61.889	0.086	1	237

(b)

	Vertical Accuracy - RMSEZ			95% Conf. Level		GCPs	Proc. Time
	avg (m)	std dev (m)	skew	kurtosis	V (m)	V	(min)
Mean	0.026	0.018	1.055	2.026	0.051	62.9	111.9
SD	0.002	0.003	0.748	4.694	0.004	0.3	81.6
Max	0.029	0.027	4.553	29.112	0.057	63	247
Min	0.019	0.014	0.312	-0.761	0.037	62	10
Range	0.010	0.013	4.241	29.873	0.020	1	237

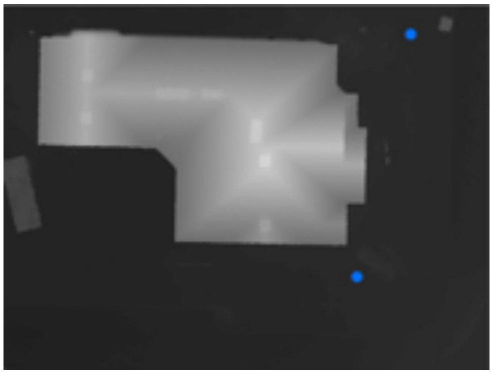


(a)

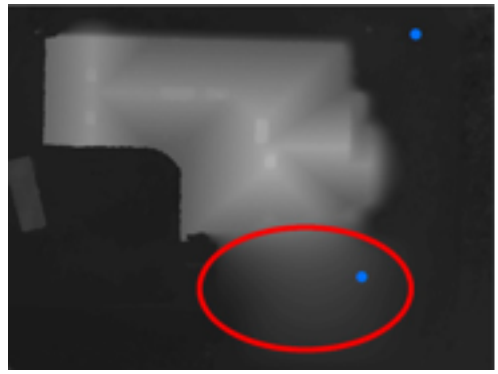


(b)

Figure 9: Scatter plot diagrams of the relationship between spatial accuracy assessment results and processing time for trials. (a) Easting, Northing, & Horizontal (b) Vertical – 48 trials & Vertical – 45 trials after removing the three outlying trials



(a)



(b)

Figure 10: Comparison of normal digital elevation model (a) and digital elevation model (b) created with 'highest accuracy' alignment, 20000 'key point limit', and 500 'tie point limit'. The bright white areas are higher elevations and the darker areas are lower elevations. The red circle shows areas where vertical checkpoints are measuring inaccurate elevations because the edges of the building roof to the east and to the south of the building and the ground are not well defined.

3.2. Geometric Quality Assessment Results

The geometric quality assessment results for selected trials are found in Table 8. A comparison between orthophotos generated with 'medium quality' dense clouds (Trial 13 & Trial 25) and 'high quality' dense clouds was conducted first to analyze the difference in image quality. The R-squared values derived from the linear regressions on the left side of Figure 11 offer statistical support that the shorter processing times from the 'medium quality' dense clouds have a direct correspondence to an increase in the presence of both image artifacts and image blur for the same set of randomly generated circles. The negative coefficients and p-values support the significant correspondence that a decrease in processing time led to an increase in image artifacts (coefficient = -0.070; p = 0.007), image blur (coefficient = -0.058; p = 0.026), and data voids (coefficient = -0.017; p = 0.106).

Thus to preserve the highest image quality, the authors decided to compare only 'high quality' dense clouds based on different 'accuracy' alignments by removing 'low' or 'medium quality' generated dense clouds from consideration. This meant that many of the fastest trial processing times were removed from consideration going forward. Trials 30, 32, 36, 37, 38, and 40 were compared to each other in Figure 11. Differences in image artifact and image blur presence between the three alignment types over the six trials relative to processing time were not significant as shown by the low R-squared values on the right-hand side of Figure 11. Further, the subsequent simple linear regressions statistically supported this lack of significance for image artifacts (p = 0.753), image blur (p = 0.203), and data voids (p = 0.300). Data voids were highest for the 'lowest accuracy' alignment trials, 30 and 32.

Table 8: Geometric quality assessment results for selected trials

#	Alignment 'accuracy'	Key pt. limit	Key pt limit	Dense cloud 'quality'	N	Image Artifacts	Image Blur	Data Voids	Proc. Time (min)
13	Medium	20000	500	Medium	100	45	42	16	49
25	Medium	40000	1000	Medium	100	46	35	13	79
30	Lowest	4000	2500	High	100	38	30	15	141
32	Lowest	8000	5000	High	100	30	25	14	143
33	Medium	20000	500	High	100	34	26	12	185
36	Medium	4000	2500	High	100	37	31	11	200
37	Low	4000	2500	High	100	33	27	10	201
38	Low	4000	5000	High	100	33	28	15	211
40	Medium	8000	5000	High	100	31	27	12	214
44	Medium	40000	1000	High	100	35	32	13	235

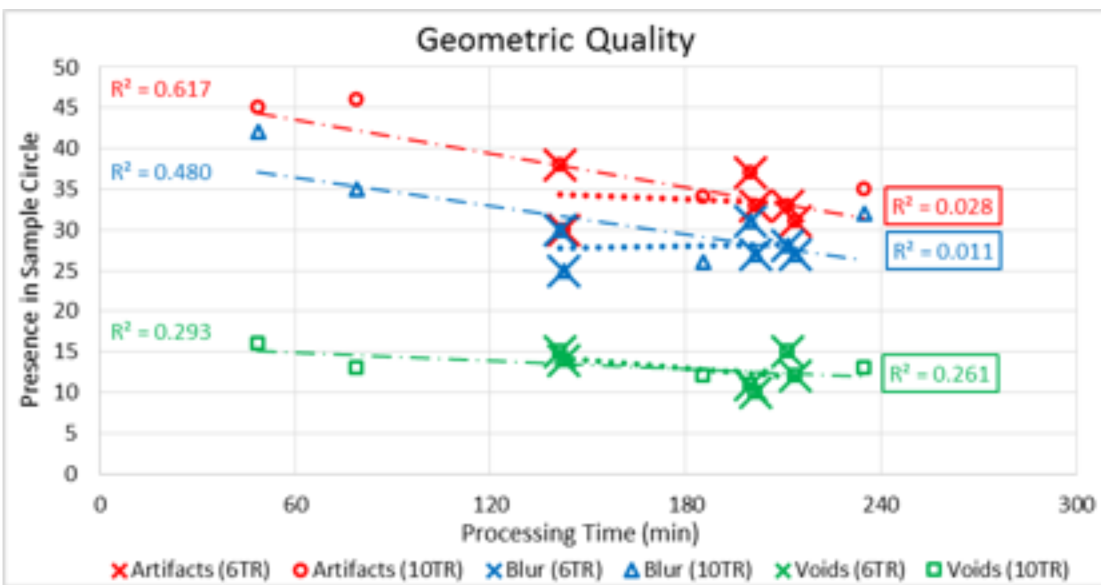


Figure 11: Geometric quality assessment of orthophotos. (a) The dash-dot linear trend lines indicate a direct relationship between presence of image artifacts and processing time for the ten selected trials that include both 'medium' and 'high quality' dense clouds. The respective R-squared values for these dash-dot trend lines are on the left side of the figure. The dot-only linear trend lines indicate the variation in image artifacts and image blur does not correspond to processing time for the six remaining trials after removing the 'medium quality' dense clouds and large 'key point limit' trials. The corresponding R-squared values for these trend lines are on the right side of the figure and outlined.

4. Discussion

Changing the ‘accuracy’ alignment parameter, the ‘key point limit’, the ’tie point limit’, and the generated dense cloud ‘quality’ parameter provided a wide range of processing times for the generation of the digital elevation model and the orthophoto mosaic. The total processing time for these 48 trials ranged from 10 minutes to 247 minutes (4 hours, 7 minutes). With 122 images used for this study area, total processing time on a per image basis ranged from 5 s/image to 121 s/image. This wide range in processing times yielded minimal changes in either the horizontal or vertical spatial accuracy. The statistical relationship between processing time and spatial accuracy was either significant with weak negative correlation (horizontal) or insignificant (vertical). Thus, substantial time savings can be had if the users’ primary objective is to obtain spatially accurate orthophotos and coarse resolution digital elevation models.

The final alignment ‘accuracy’ parameter and the dense cloud generation ‘quality’ parameter relate to the spatial resolution of the imagery used in the UAS-MVS processing. The range in processing times across the 48 trials supports the direct correspondence in reduced processing time derived from the selection of lower image resolution parameters. Interestingly, the use of higher fidelity image resolution during alignment does not translate to improvements in spatial accuracy.

While ‘high quality’ dense cloud generation takes longer to process than ‘low’ or ‘medium quality’ dense cloud generation, the geometric quality assessment showed that orthophoto mosaic quality is significantly better when using a ‘high quality’ generated dense cloud. Therefore, it is recommended that processing is completed using a ‘high quality’ dense cloud to minimize image artifacts and image blur. Given slight improvements in spatial accuracy using a more traditional ratio of key points to tie points and slightly better data completeness in the visual accuracy assessment (i.e., fewer voids), the optimal settings derived from the 48 trials for a low to moderate topographic relief study area are from Trial 37 and shown in Table 9.

Table 9: Final Photoscan processing parameters for alignment and dense point cloud generation derived from spatial accuracy assessment and geometric quality assessment.

Alignment	
Accuracy	Low
Pair preselection	Disabled
Key point limit	4000
Tie point limit	2500
Constrain features by mask	No
Dense Point Cloud Generation	
Quality	High
Filtering mode	Moderate

Discussion of these results within a broader context of sUAS mapping is critical. Broader contextual considerations include project site characteristics, data acquisition methods, computer resources, software processing, and applications across a variety of disciplines. The remainder of the discussion is split into two sections based on implications for field operations and data processing across multiple disciplines.

4.1. Field operations

Site characteristics and data acquisition techniques have implications on field operation procedures and efficiency. While multiple site variables can potentially affect UAS-derived geospatial datasets, the dominant characteristics of a site are its topography and its land cover. The study area featured herein and shown in Figure 2 is a residential area with low to moderate topographic relief. It contains a variety of land cover including both vegetated areas (e.g., grass, shrubs, dense tree canopy, sparse tree canopy) and improved areas (e.g., roads, structures). Thus, the study area is representative of common land development parcels here in the southeastern United States and other areas worldwide. With the adoption of Part 107 regulations in the United States (FAA, 2016), there is rapid growth in commercial UAS operations mapping existing infrastructure and construction activities. This study provides applicable results that operators can use with similar site characteristics.

Across the disciplines that have adopted UAS mapping technology, certain site characteristics are more prevalent than others within a given discipline. For resource extraction applications such as open pit mining where UAS are used to map removal of material from mine walls or volumetric computation of piles (Chen, Li, Chang, Sofia, & Tarolli, 2015; Tong et al., 2015), substantial topographic relief is common and expected. These areas with sharp changes in elevation and extreme topographic gradients may be more susceptible to errors in vertical accuracy when using digital elevation models generated from less rigorous processing parameters (e.g., 'low quality' dense clouds). In forestry applications, vegetated land cover and tree canopy density play significant roles in obscuring the ground surface for accurate elevation surface modeling during leaf on conditions (Dandois & Ellis, 2013). However, there is inherent variability in site characteristics within disciplines as well. For example, within the subdiscipline of coastal geomorphology, substantial topographic relief is common in cliff surveying (M. R. James & Robson, 2012; Westoby, Brasington, Glasser, Hambrey, & Reynolds, 2012) while low and moderate topographic relief is prevalent in beach monitoring (Casella et al., 2016; Gonçalves & Henriques, 2015). Given this variability in site characteristics within and across disciplines, additional investigation into the tradeoffs between processing time, spatial accuracy, and geometric quality is advised across a variety of topographic gradients and land cover types to avoid misapplication of the results reported herein.

Appropriate data acquisition techniques are critical to achieving optimal spatial accuracies of resultant UAS-derived geospatial datasets. Data acquisition is broadly categorized as either control or image acquisition. In this study, ground control was collected by precise measurement using a robotic total station. While total station surveying is a more accurate technique than the establishment of either ground or airborne control by GNSS surveying alone (Ghilani & Wolf, 2014), total station surveying is a more time intensive process over large survey areas with airborne GNSS being the least time intensive surveying technique. When considering the tradeoffs between field time and desired spatial accuracy, the accuracy of the input control is critical to the reproducibility and subsequent accuracy of the derived SfM datasets (Clapuyt et al., 2016). With input vertical and horizontal accuracies of less than 1 cm for both the ground control and checkpoints, this study reinforces the concept that the resultant horizontal and vertical geospatial dataset spatial accuracy is limited by the accuracy of the control (Toby N. Tonkin & Midgley, 2016). As mentioned earlier, site characteristics (e.g., closed canopies in forestry applications) can limit both the methods of control establishment as well as the upper threshold of expected spatial accuracy.

Both hardware (i.e., cameras) and flight planning operations can affect image acquisition and the subsequent spatial accuracy and geometric quality of UAS-derived geospatial datasets. In this study, the UAS platform flew at a consistent 55m AGL capturing near-nadir perspective imagery with a Sony A6000 non-metric camera. For non-metric cameras where self-calibration is needed, James et al. (2014) found that collecting convergent imagery could reduce doming error effects. Carbonneau et al. (2016) completed additional research with multiple non-metric cameras showing the cost effectiveness of collecting both nadir and oblique imagery at multiple altitudes in improving camera self-calibration parameters. The authors did not collect convergent imagery as part of this data acquisition but do consider this flight acquisition method (Carbonneau & Dietrich, 2016; Mike R. James & Robson, 2014) to be a worthy endeavor especially at the entry level end of the UAS market where camera self-calibration is required. Investigation into the impact that convergent imagery has on processing time and alignment processing time in particular is recommended in subsequent study. Further, considerable work related to camera self-calibration and UAS flight planning parameters has been completed within the geomorphology community thus far (Carbonneau & Dietrich, 2016; M. R. James et al., 2017; Mike R. James & Robson, 2014). Additional research into the impact flight planning characteristics have on the spatial accuracy and geometric quality of UAS-derived geospatial datasets in non-geomorphological applications (e.g., agriculture, forestry) is encouraged as well.

JOURNAL OF UNMANNED AERIAL SYSTEMS

4.2. Data Processing

The computer hardware configuration described in Section 2.4 is representative of a typical desktop workstation that exceeds the minimum specifications set out in the Agisoft Photoscan manual (Agisoft, 2016). For each activated GPU, one CPU was subsequently deactivated per the manual. The GPUs are critical to processing time improvement due to Agisoft support of OpenCL acceleration during the depth maps reconstruction phase of building dense point clouds. The building dense clouds step is the most time intensive step as shown in Table 2. To estimate the impact that GPUs have on processing time during dense cloud generation, the authors conducted a small test using the parameters found in Table 9 with eight CPUs and zero GPUs activated (C8G0) versus seven CPUs and one GPU activated (C7G1). The total processing time and depth map generation processing time for C8G0 was 5:44:00 hrs and 5:00:00 hrs, respectively. Meanwhile, the total processing time and depth map generation processing time for C8G0 was 3:22:00 hrs and 2:35:00 hrs, respectively. On a per image basis (122 images total), the use of a CPU instead of a GPU resulted in an increase of 72 s/image in depth map generation processing time. This result highlights the importance of using at least one GPU to enhance processing time as supported by the literature (Agisoft, 2016; Verhoeven, 2011). Without access to more than one GPU, it is hypothesized that the depth map generation processing time as part of the dense point cloud generation step could be decreased with the substitution of additional GPUs for CPUs. Further investigation is needed to confirm the magnitude of the hypothesized decrease both within Photoscan and other SfM software packages that use GPUs (e.g., Pix4D (Pix4D, 2016), VisualSFM (Wu, 2016)).

Cloud and cluster computing continue to grow as options for processing large amounts of geospatial data (Li, Yang, Liu, Hu, & Jin, 2016; Sugumaran, Hegeman, Sardeshmukh, & Armstrong, 2015; Yang, Raskin, Goodchild, & Gahegan, 2010). However, internet accessibility, internet quality, and even finances can all hamper the availability of these robust computing resources for some operators and applications. For example, using UAS-derived orthophotos to delineate property boundaries in remote areas through participatory mapping requires a quick turnaround time from field collection to community-based exercises (Barnes & Volkmann, 2015). Data processing in remote areas is often without the benefit of high bandwidth internet for accessing cloud computing solutions (Turner et al., 2014). Thus, a mobile desktop or a laptop must be used to process the data. Ensuring that these mobile computer resources have at least one GPU is imperative for processing time efficiency.

For SfM software, this study exclusively examined Photoscan due to its wide adoption, extensive user community, affordable purchase price, and reporting features as outlined in Section 2.4. While further investigation of SfM processing time across multiple software packages is encouraged, literature (Turner et al., 2014) suggests that processing time using similar parameters is relatively the same between the two predominant SfM commercial software packages, Agisoft Photoscan (Agisoft, 2016) and Pix4D (Pix4D, 2016). The exact parameters recommended in Table 9 cannot be directly transferred to other software packages due to slightly different algorithms and implementations. The overarching results of this study are that image downscaling during alignment and dense cloud generation can yield significant time efficiency gains without sacrificing spatial accuracy in low to moderate topographic relief project sites. This broader result is reasonably transferrable based on the recent literature (Gross & Heumann, 2016) finding no statistically significant difference in spatial accuracy between the predominant SfM software packages.

As shown in Table 2, the overall processing time is primarily a result of two processing steps: the final alignment and the dense cloud generation parameters. While not nearly as time intensive as dense cloud generation, the final alignment step can substantially affect processing time depending upon the parameters selected. In this study, changing the alignment ‘accuracy’ of the default setting from ‘highest’ to ‘lowest’ resulted in a total time decrease from 0:50:20 hrs (‘highest accuracy’) to 0:02:39 hrs (‘lowest accuracy’) or a decrease in processing time of 23 s/image. With minimal variation in the horizontal and vertical spatial accuracy across all trials, the alignment parameters (e.g., ‘accuracy’, ‘keypoint limit’, ‘tiepoint limit’) had nominal impact on spatial accuracy for this study area.

The EXIF headers of the images used in this study lacked georeferencing information. Therefore, the ‘pair preselection’ parameter was disabled in Table 9. Consequently, the Photoscan software did not have any preconditions to assist with imagery alignment. This ‘disabled’ setting could be considered the worst case scenario for imagery alignment. Meanwhile, the best case scenario for imagery alignment with regards to ‘pair preselection’ is a ‘reference’ setting that uses precise camera exposure stations derived from phase-based differential GNSS processing. Using the settings found in Table 9 and an accurate post-processed kinematic (PPK) GNSS navigation trajectory of the camera exposure stations for this same study site, the authors found that changing to a ‘reference pair preselection’ alignment parameter yielded a decrease in processing time of 0.5 s/image. Likewise, changing the ‘accuracy’ parameter in Table 9 from ‘low’ to ‘medium’ (i.e., less image downscaling) and using ‘reference pair preselection’ yielded a decrease in processing time of 1 s/image for the PPK GNSS camera exposure positions relative to a ‘disabled’ setting. For a dataset with a larger number of images, this time savings could prove beneficial; however, for this size project area, the time difference is small relative to other parameter settings. The real benefits of using precise airborne GNSS control are realized through time savings in the field (M. R. James et al., 2017) and for applications such as forestry where the placement of unobscured ground control is difficult (Dandois & Ellis, 2013). While differential GNSS positioning using carrier phase observations is gaining traction in the UAS community, most entry level UAS rely on code-based positioning (Carbonneau & Dietrich, 2016). The processing time benefits of using approximate non-differential, code-based GNSS camera positions fall between the precise PPK GNSS derived camera positions tested above and the lack of alignment preconditioning used in the overall study.

The most time intensive step in the SfM software processing is the dense point cloud generation as shown in Table 2. The authors varied the dense point cloud generation ‘quality’ parameter to test the effect image downscaling had on spatial accuracy, geometric quality, and processing time. Across all trials, the minimal variation in spatial accuracy suggests that image downscaling during dense point cloud construction had a marginal effect on the spatial accuracy of generated orthophotos and digital elevation models. Altering the ‘quality’ parameter yielded significant differences in processing time. The ‘ultra high quality’ setting was dismissed as an optimal setting since the dense cloud generation processing time was about 570 s/image for this study area. In comparison, the ‘high’, ‘medium’, and ‘low’ settings contributed an average dense cloud generation processing time across all trials of 86 s/image, 17 s/image, and 4 s/image, respectively. Therefore, it is clear that there are significant processing time gains that can be made from downscaling the input imagery at this step. Areas with sharp changes in elevation and extreme topographic gradients may require more stringent dense cloud generation parameters to maintain a level of spatial accuracy that is nearly equivalent to the accuracy of the input georeferencing control. Thus, agricultural applications with relatively flat topography (Caturegli et al., 2016) would be less likely to be impacted by the dense point clouds generated with ‘low quality’ settings than geomorphological applications in steeper terrain (M. R. James & Robson, 2012; Warrick, Ritchie, Adelman, Adelman, & Limber, 2016; Westoby et al., 2012) or with hummocky relief (Toby N. Tonkin & Midgley, 2016). Further investigation is needed with project sites in these steeper terrain areas to quantify the susceptibility of these areas to errors in vertical accuracy when using digital elevation models generated from lower density point clouds.

The results from the geometric quality assessment showed that there were significant reductions in image artifacts, image blur, and data voids when using ‘high quality’ dense cloud generation parameters instead of ‘medium quality’. Thus, the user needs to determine the tradeoff between enhanced qualitative fidelity of the derived geospatial datasets in terms of distinguishing and identifying features within the imagery versus the benefits of reduced processing time. For example, in disaster response applications (Pajares, 2015), faster processing times resulting in geospatial datasets with poorer spatial accuracies and less qualitative fidelity are more acceptable when the mapping products are used to quickly and effectively determine which houses are damaged and what areas might need immediate assistance. In forestry and ecological applications, the difference between classifying land cover by general classes versus specific species can lie in the qualitative fidelity of the resultant mapping products. For other applications such as infrastructure surveying (Colomina & Molina, 2014), higher accuracy mapping products with greater image fidelity may be needed for roadway design and construction. As with any UAS mapping project, client and end user needs dictate the benchmarks to set for spatial accuracy and geometric quality of the final geospatial datasets. The data acquisition techniques (i.e., control and image acquisition) and the rigorousness of the SfM processing parameters need to be tailored by the operator to meet these benchmarks.

5. Conclusions

This study sought a more time efficient Structure from Motion (SfM) processing approach for generating unmanned aerial system (UAS) derived geospatial datasets, specifically orthophoto mosaics and digital elevation models. To analyze the impact that SfM processing parameter selection had on spatial accuracy, geometric quality, and processing time, the representative SfM software package for image processing was Agisoft Photoscan v1.2.5. A MAP-M4 multirotor quadcopter with a 16mm Sony a6000 camera captured 122, ~1.25 cm/pixel ground resolution images of a study area in Alachua County, FL, USA. Spatial accuracy (i.e., horizontal and vertical difference between ground truth data and the final mapping products) and geometric quality (i.e., qualitative fidelity of the derived orthophoto mosaics in terms of distinguishing and identifying features within the imagery) were the primary characteristics evaluated in determining the tradeoff between processing time and the geospatial dataset accuracy and quality.

In low to moderate topographic relief locations, the results indicate that less rigorous Photoscan processing parameters, specifically alignment and dense cloud generation parameters, provide substantial time savings without sacrificing the spatial accuracy of UAS-derived geospatial datasets. Lower 'quality' parameters in the dense cloud building step of the UAS mapping workflow yield the greatest time savings. The 'high', 'medium', and 'low' 'quality' settings contributed an average dense cloud generation processing time across all trials of 86 s/image, 17 s/image, and 4 s/image, respectively. For projects with a large number of images, the impact of image downscaling at this step leads to significant processing time improvement. When considering geometric quality in addition to spatial accuracy, reductions in the alignment 'accuracy' and the 'key point limit' did not impact the spatial accuracy of the resultant geospatial datasets. To further enhance understanding of tradeoffs between processing time, spatial accuracy, and geometric quality of UAS derived geospatial datasets, recommendations for future work include applying the procedures and results found herein across a variety of both SfM software packages (e.g., Pix4D) and project sites (e.g., more extreme topographic gradients, differing land cover).

Acknowledgments

Acknowledgments: This is a collaborative study between the University of Florida (UF) Geomatics program and Micro Aerial Projects L.L.C. (http://www.microaerialprojects.com/). No external sources of funding were used to conduct this study. Thank you to Oliver Volkmann and Nicholas DiGruttolo for field data collection and UAS operations. Thank you to Levente Juhasz for assistance with automating data extraction.

REFERENCES

Agisoft. (2016). Agisoft PhotoScan User Manual: Professonal Edition, Version 1.2. Agisoft LLC. Retrieved from http://www.agisoft.com/pdf/photoscan-pro_1_2_en.pdf

ASPRS. (1990, March). ASPRS Positional Accuracy Standards for Digital Geospatial Data. American Society for Photogrammetry and Remote Sensing. Retrieved from http://www.asprs.org/a/society/committees/standards/1990_jul_1068-1070.pdf

ASPRS. (2015). ASPRS Positional Accuracy Standards for Digital Geospatial Data. Photogrammetric Engineering & Remote Sensing, 81(3), 53.

Barnes, G., & Volkmann, W. (2015). High-Resolution Mapping with Unmanned Aerial Systems. Surveying and Land Information Science, 74(1), 5–13.

CAA. (2016). The Air Navigation Order 2016 and Regulations (FAA News No. CAP 393) (p. 408). London, UK: Civil Aviation Authority. Retrieved from http://publicapps.caa.co.uk/docs/33/CAP393Ed5Am1_OCT2016_BM.pdf

Carbonneau, P. E., & Dietrich, J. T. (2016). Cost-effective non-metric photogrammetry from consumer-grade sUAS: implications for direct georeferencing of structure from motion photogrammetry. Earth Surface Processes and Landforms, n/a-n/a. https://doi.org/10.1002/esp.4012

Casella, E., Rovere, A., Pedroncini, A., Stark, C. P., Casella, M., Ferrari, M., & Firpo, M. (2016). Drones as tools for monitoring beach topography changes in the Ligurian Sea (NW Mediterranean). Geo-Marine Letters, 36(2), 151–163. https://doi.org/10.1007/s00367-016-0435-9

Caturegli, L., Corniglia, M., Gaetani, M., Grossi, N., Magni, S., Migliazzi, M., … Volterrani, M. (2016). Unmanned Aerial Vehicle to Estimate Nitrogen Status of Turfgrasses. PLOS ONE, 11(6), e0158268. https://doi.org/10.1371/journal.pone.0158268

Chen, J., Li, K., Chang, K.-J., Sofia, G., & Tarolli, P. (2015). Open-pit mining geomorphic feature characterisation. International Journal of Applied Earth Observation and Geoinformation, 42, 76–86. https://doi.org/10.1016/j.jag.2015.05.001

Clapuyt, F., Vanacker, V., & Van Oost, K. (2016). Reproducibility of UAV-based earth topography reconstructions based on Structure-from-Motion algorithms. Geomorphology, 260, 4–15. https://doi.org/10.1016/j.geomorph.2015.05.011

Colomina, I., & Molina, P. (2014). Unmanned aerial systems for photogrammetry and remote sensing: A review. ISPRS Journal of Photogrammetry and Remote Sensing, 92, 79–97. https://doi.org/10.1016/j.isprsjprs.2014.02.013

Dandois, J. P., & Ellis, E. C. (2013). High spatial resolution three-dimensional mapping of vegetation spectral dynamics using computer vision. Remote Sensing of Environment, 136, 259–276. https://doi.org/10.1016/j.rse.2013.04.005

Dietrich, J. T. (2016). Riverscape mapping with helicopter-based Structure-from-Motion photogrammetry. Geomorphology, 252, 144–157. https://doi.org/10.1016/j.geomorph.2015.05.008

ESRI. (2016). ArcGIS 10.4 for Desktop (ArcMap). Environmental Systems Research Institute, Inc. Retrieved from https://desktop.arcgis.com/en/arcmap/10.4/

FAA. (2016). Summary of small unmanned aircraft rule (Part 107) (FAA News) (p. 3). Washington, DC: Federal Aviation Administration. Retrieved from https://www.faa.gov/uas/media/Part_107_Summary.pdf

FGDC. (1998). Geospatial Positioning Accuracy Standards, Part 3: National Standard for Spatial Data Accuracy (NSSDA) (No. FGDC-STD-007.3-1998.) (p. 28). Reston, VA: Federal Geographic Data Committee. Retrieved from https://www.fgdc.gov/standards/projects/accuracy/part3/chapter3

Flood, M. (2004). Vertical accuracy reporting for LiDAR data. American Society for Photogrammetry and Remote Sensing LiDAR Committee.

Fonstad, M. A., Dietrich, J. T., Courville, B. C., Jensen, J. L., & Carbonneau, P. E. (2013). Topographic structure from motion: a new development in photogrammetric measurement. Earth Surface Processes and Landforms, 38(4), 421–430. https://doi.org/10.1002/esp.3366

Gerke, M., & Przybilla, H.-J. (2016). Accuracy Analysis of Photogrammetric UAV Image Blocks: Influence of Onboard RTK-GNSS and Cross Flight Patterns. Photogrammetrie - Fernerkundung - Geoinformation, 2016(1), 17–30. https://doi.org/10.1127/pfg/2016/0284

Ghilani, C. D., & Wolf, P. R. (2014). Elementary Surveying: An Introduction to Geomatics (14th ed.). Upper Saddle River, N.J: Pearson Education, Inc.

Gonçalves, J. A., & Henriques, R. (2015). UAV photogrammetry for topographic monitoring of coastal areas. ISPRS Journal of Photogrammetry and Remote Sensing, 104, 101–111. https://doi.org/10.1016/j.isprsjprs.2015.02.009

Gross, J. W., & Heumann, B. W. (2016). A Statistical Examination of Image Stitching Software Packages For Use With Unmanned Aerial Systems. Photogrammetric Engineering & Remote Sensing, 82(6), 419–425. https://doi.org/10.14358/PERS.82.6.419

Harwin, S., & Lucieer, A. (2012). Assessing the Accuracy of Georeferenced Point Clouds Produced via Multi-View Stereopsis from Unmanned Aerial Vehicle (UAV) Imagery. Remote Sensing, 4(6), 1573–1599. https://doi.org/10.3390/rs4061573

James, M. R., & Robson, S. (2012). Straightforward reconstruction of 3D surfaces and topography with a camera: Accuracy and geoscience application. Journal of Geophysical Research: Earth Surface, 117(F3), F03017. https://doi.org/10.1029/2011JF002289

James, M. R., Robson, S., d'Oleire-Oltmanns, S., & Niethammer, U. (2017). Optimising UAV topographic surveys processed with structure-from-motion: Ground control quality, quantity and bundle adjustment. Geomorphology, 280, 51–66. https://doi.org/10.1016/j.geomorph.2016.11.021

James, Mike R., & Robson, S. (2014). Mitigating systematic error in topographic models derived from UAV and ground-based image networks. Earth Surface Processes and Landforms, 39(10), 1413–1420. https://doi.org/10.1002/esp.3609

Javernick, L., Brasington, J., & Caruso, B. (2014). Modeling the topography of shallow braided rivers using Structure-from-Motion photogrammetry. Geomorphology, 213, 166–182. https://doi.org/10.1016/j.geomorph.2014.01.006

Li, Z., Yang, C., Liu, K., Hu, F., & Jin, B. (2016). Automatic Scaling Hadoop in the Cloud for Efficient Process of Big Geospatial Data. ISPRS International Journal of Geo-Information, 5(10), 173. https://doi.org/10.3390/ijgi5100173

Messinger, M., Asner, G., & Silman, M. (2016). Rapid Assessments of Amazon Forest Structure and Biomass Using Small Unmanned Aerial Systems. Remote Sensing, 8(8), 615. https://doi.org/10.3390/rs8080615

Pajares, G. (2015). Overview and Current Status of Remote Sensing Applications Based on Unmanned Aerial Vehicles (UAVs). Photogrammetric Engineering & Remote Sensing, 81(4), 281–330. https://doi.org/10.14358/PERS.81.4.281

Paneque-Gálvez, J., McCall, M., Napoletano, B., Wich, S., & Koh, L. (2014). Small Drones for Community-Based Forest Monitoring: An Assessment of Their Feasibility and Potential in Tropical Areas. Forests, 5(6), 1481–1507. https://doi.org/10.3390/f5061481

Pix4D. (2016). Pix4D mapper 2.2 Manual. Retrieved September 20, 2016, from https://support.pix4d.com

Puliti, S., Olerka, H., Gobakken, T., & Næsset, E. (2015). Inventory of Small Forest Areas Using an Unmanned Aerial System. Remote Sensing, 7(8), 9632–9654. https://doi.org/10.3390/rs70809632

RTKLIB. (2017). RTKLIB: An Open Source Program Package for GNSS Positioning. Retrieved March 27, 2017, from http://www.rtklib.com/rtklib.htm

Sauerbier, M., Siegrist, E., Eisenbeiss, H., & Demir, N. (2011). The practical application of UAV-based photogrammetry under economic aspects. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 38(1). Retrieved from http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XXXVIII-1-C22/45/2011/isprsarchives-XXXVIII-1-C22-45-2011.pdf

Sugumaran, R., Hegeman, J. W., Sardeshmukh, V. B., & Armstrong, M. P. (2015). Processing remote-sensing data in cloud computing environments. In Remotely Sensed Data Characterization, Classification, and Accuracies (pp. 553–562). CRC Press. Retrieved from http://www.crcnetbase.com/doi/pdf/10.1201/b19294-38

Tavani, S., Granado, P., Corradetti, A., Girundo, M., Iannace, A., Arbués, P., … Mazzoli, S. (2014). Building a virtual outcrop, extracting geological information from it, and sharing the results in Google Earth via OpenPlot and Photoscan: An example from the Khaviz Anticline (Iran). Computers & Geosciences, 63, 44–53. https://doi.org/10.1016/j.cageo.2013.10.013

Tong, X., Liu, X., Chen, P., Liu, S., Luan, K., Li, L., … Hong, Z. (2015). Integration of UAV-Based Photogrammetry and Terrestrial Laser Scanning for the Three-Dimensional Mapping and Monitoring of Open-Pit Mine Areas. Remote Sensing, 7(6), 6635–6662. https://doi.org/10.3390/rs70606635

Tonkin, T. N., Midgley, N. G., Graham, D. J., & Labadz, J. C. (2014). The potential of small unmanned aircraft systems and structure-from-motion for topographic surveys: A test of emerging integrated approaches at Cwm Idwal, North Wales. Geomorphology, 226, 35–43. https://doi.org/10.1016/j.geomorph.2014.07.021

Tonkin, Toby N., & Midgley, N. G. (2016). Ground-Control Networks for Image Based Surface Reconstruction: An Investigation of Optimum Survey Designs Using UAV Derived Imagery and Structure-from-Motion Photogrammetry. Remote Sensing, 8(9), 786. https://doi.org/10.3390/rs8090786

Turner, D., Lucieer, A., & Wallace, L. (2014). Direct Georeferencing of Ultrahigh-Resolution UAV Imagery. IEEE Transactions on Geoscience and Remote Sensing, 52(5), 2738–2745. https://doi.org/10.1109/TGRS.2013.2265295

Verhoeven, G. (2011). Taking computer vision aloft – archaeological three-dimensional reconstructions from aerial photographs with photoscan. Archaeological Prospection, 18(1), 67–73. https://doi.org/10.1002/arp.399

Wallace, L., Lucieer, A., Malenovský, Z., Turner, D., & Vopěnka, P. (2016). Assessment of Forest Structure Using Two UAV Techniques: A Comparison of Airborne Laser Scanning and Structure from Motion (SfM) Point Clouds. Forests, 7(3), 62. https://doi.org/10.3390/f7030062

Warrick, J. A., Ritchie, A. C., Adelman, G., Adelman, K., & Limber, P. W. (2016). New Techniques to Measure Cliff Change from Historical Oblique Aerial Photographs and Structure-from-Motion Photogrammetry. Journal of Coastal Research. Retrieved from http://www.jcronline.org/doi/abs/10.2112/JCOASTRES-D-16-00095.1

Westoby, M. J., Brasington, J., Glasser, N. F., Hambrey, M. J., & Reynolds, J. M. (2012). "Structure-from-Motion" photogrammetry: A low-cost, effective tool for geoscience applications. Geomorphology, 179, 300–314. https://doi.org/10.1016/j.geomorph.2012.08.021

Wolf, P. R., & Dewitt, B. A. (2000). Elements of Photogrammetry : with Applications in GIS. Boston: McGraw-Hill.

Wu, C. (2016, September 20). VisualSFM : A Visual Structure from Motion System. Retrieved from http://ccwu.me/vsfm/

Yang, C., Raskin, R., Goodchild, M., & Gahegan, M. (2010). Geospatial Cyberinfrastructure: Past, present and future. Computers, Environment and Urban Systems, 34(4), 264–277. https://doi.org/10.1016/j.compenvurbsys.2010.04.001

JOURNAL OF UNMANNED AERIAL SYSTEMS

PEER-REVIEWED ARTICLE

QUALITATIVE COMPARISON OF THE SLANTED TAILPLANE CONFIGURATION IN THE INDONESIAN BPPT'S GENERAL PURPOSES UAV

Ir. Jemie Muliadi, MT.
Senior Engineer, Agency for the Assessment and Application of Technology (BPPT), Indonesia

ABSTRACT

This article contains a discussion of the advantages of the unique configuration of the Indonesian general purposes Unmanned Aerial Vehicles (UAV) equipped with the slanted tailplane. These UAVs—prototyped by the Indonesian Agency for the Assessment and Application of Technology (BPPT)—applied the V-tail and the Inverted V-tail plane configuration for their empennage construction. Such unconventional configuration has been implemented in the aircraft design, especially for the UAV. Previously, the conventional configurations are preferable due to the less of resource, lower effort and lower cost in the aircraft design. In the conventional empennage design, the tail consists of the vertical fin and horizontal tail plane joined to form the T-tail configuration. Although the vertical and horizontal tail construction decoupled the aircraft longitudinal and directional control, there is a higher risk of flutter occurrence at the T-tail joint. The conventional T-tail configuration also resulted in higher drag compared with the slanted tailplane, i.e. the V and the inverted V-tail plane. Since the UAV relied on the autopilot and autonomous flight features, thus a complex control mechanism of the slanted tailplane becomes an insignificant problem compared to the drag reduction obtained. This drag reduction significantly leads to the maximum range or endurance achievement in the UAV remote missions. Besides the benefit of the unconventional tail configuration, the wing and engine placement also brought practical advantages for UAV operations, for example, to impose special flight characteristics like flight agility or the Short Take-Off/Landing (STOL) and supporting the engine maintenance. Thus, the BPPT successfully prototyped a V-Tail UAV named the BPPT-01B "Gagak" (means: the crow) and an inverted V-Tail UAV named BPPT-02A "Pelatuk (means: the woodpecker). Each of them equipped with several opposing constructions intended for the comparative research at the later stage such as the low-high wing placement and the low-high tail boom placement. The analysis of the qualitative comparison of this uniqueness shown that these slanted tailplane type will play a tremendous role in the future configuration of UAV.

Keywords: butterfly tail, UAV design, inverted V-tail, PUNA BPPT, V-tail

THE UAV DESIGN: TOWARD A NEW PARADIGM OF AIRCRAFT CONFIGURATION

The aircraft design elaborates several different fields of expertise and settled with a compromise between those fields' objectives, criterion and limitation. Stretched wider than just transporting passengers, the aircraft design has advanced to adapt the aerial mission and many similar roles. These purposes become the early motivation in the Unmanned Aerial Vehicles (UAV) development to eliminate the risk and reduce the operating cost for the aerial missions. An example of Indonesian UAV presented in Figure 1 that developed by the Indonesian Agency for The Assessment and Application of Technology (BPPT).

Figure 1. "Pelatuk" or Woodpecker, an Indonesian UAV developed by the BPPT.

In a detailed description, the aircraft design is an interdisciplinary activity that simultaneously considers the complexities of its systems, functions and requirements that are interrelated one another. The aircraft design is generally divided into the conceptual design phase, the preliminary design phase and the final design phase. Around of the 80% of the life cycle cost of an aircraft are decided in the conceptual design phase (Rizzi, 2011). This critical situation drove the designers to be carefully avoiding the mistakes in this phase since it might cost very expensive reparation in the future. There are arguments that highlight the tremendous cost and risk associated with a new technology. These factors are making the innovation become difficult for the business case, especially in the development of large transport aircraft. Most of the customer prefers the low-risk incremental improvements until they find that new ideas are well-proven, (Kroo, 2004). Consequently, aircraft designers preferred to combine as much of the conventional items that are well-validated or well-studied to defined a rough configuration of the designed aircraft.

Most of the engineering tools in the aircraft design came from the handbook methods or stemmed from the linear fluid mechanics assumptions (Rizzi, 2011). Applying the linear fluid mechanics assumptions allowed the aircraft designer in gaining the low-cost reliable aerodynamic data while such reliability is valid if the configurations are kept being conventional. Since the unconventional configuration needed the accurate description of the non-linear aspect of the aircraft flight characteristics (Rizzi, 2011), these conditions lead to the preferences of using the conventional forms over the unconventional ones. The conceptual designs also become conservative by limiting the aircraft working points in the range of its known performances (Filippone, 2000). However, the UAV in recent times has been developed in the unique and conventional configuration while such development continuously enriching the aircraft design paradigms.

Unmanned Aerial Vehicles (UAV) is recognized in various of terms such as drones, Remotely Piloted Vehicle (RPV), Unindividuated Aerial Vehicle (Jenie, 2007), or in more holistic terms i.e. Unmanned Aerial Systems (UAS). These terms referred to the collaboration between the specially designed aircraft; the special purpose sensor for the remote missions; the data link transmission between the aircraft on-board systems and its Ground Control Station (GCS); the mission center and the autonomous flight technology. These technologies had stimulated the development of the methods in solving the design constraints and the requirement for fulfilling the objectives of the UAV.

The UAV development has emerged the new paradigm and perspectives in the Aircraft Design. Previously, the aircraft was piloted by the skillful certified persons onboard for transporting passengers and payloads. These conditions applied tight constraints in the aircraft designing for safety reasons, reliability, lowering the risks and increasing the passengers comfortably. Thus, the conventional designs of aircraft preserved the basic pattern of configuration. These recent aircraft look as if they are almost unchanged from their ancestors (Kroo, 2004). As seen in the previous days, most of the aircraft were consist of a cylindrical fuselage, carried in flight by a pair of rectangular or polygonal symmetric wings, then equipped by vertical tailplane and horizontal tailplane to stabilize its flight.

In the manned aircraft, obvious innovations occurred rarely at the vehicle's configuration. Thus, this apparent lack of innovation triggered some suggestion which concluding that aeronautics is a mature field (Kroo, 2004). For example, the innovation in the wing and tail configuration were applied to the supersonic transport jet (Sun & Smith, 2017) by combining the wing and horizontal tail into a common plane, known as the delta wing (Smith, 2016). This delta wing planform is a result from configuration compromise and trade-offs to ensure satisfactory flight characteristics at low speed without decreasing too much the supersonic aerodynamic performance (Chambers, 2005). The future Concorde-2 also applied this delta wing platform (Mali et al., 2016).

Another innovation is the current issue, the joined-wing configuration (Cavallaro & Demasi, 2016). Moreover, the deviation from the T-tail configuration also occurred in the Beech V-Tail Bonanza which is grounded recently due to the safety purposes (Musa et al., 2015), although there is inconclusive interpretation about the relation of the V-tail with the accidental causes (Collins, 2012). Furthermore, the innovations in the manned aircraft configurations mostly applied to the military aircraft (Grissom et al., 2016), obviously for their extensive maneuvers and dangerous mission.

As the aircraft stabilizer, the tail planes are stabilizing the longitudinal and directional mode of flight. In the conventional fashion, the tail consists of a horizontal plane and vertical fin to maintain longitudinal stability and directional stability separately. The clear separation between these tailplanes was meant to support the pilot's easiness in controlling the aircraft's pitching and yawing movement. The pitching mode is controlled by deflecting the elevator located at the trailing edge of the horizontal tailplane. In the similar fashion, the yaw is controlled by deflecting the rudder located at the trailing edge of the vertical tailplane. This is the simple flight control mechanism since the complex control system is usually not considered in the initial conceptual phases of an aircraft design (Richardson et al., 2011).

The tail planes, including the elevator and rudder, bear the stability and control of the aircraft flight to preserve the passenger's comfort in flight. Since the UAV flies with no human on-board, several requirements such as the passengers' comfortably in flight and the pilot's comfortably to control become irrelevant in the design of the unmanned aircraft configuration. By deprioritizing such criterion, the new paradigm of aircraft design has evolved in the UAV development. In order to assess this new paradigm, this article will address the qualitative exploration especially in the unconventional tail configuration of the Indonesian large UAVs. By comparing the two types of the slanted tail configuration, their advantages are enhanced and their disadvantages are also identified.

To qualitatively perform the comparison, this article will be organized into five sections as follows. This opening section had served as the introductory part to messaging the importance of the unconventional tail of UAV configuration. After narrating the brief background of the topic, objectives and the scope of this chapter, the second section will explore the current aerial mission which enhanced the development of the slanted tail UAV. Entitled as "**UAV tail configuration modification: exploring the requirements of the current aerial mission**," the second section explains the problem statements. The description of the slanted tailed UAVs itself is covered in the third section, entitled "**The UAV unconventional tail configuration**". The comparative study performed by exposing the BPPT multipurpose UAV with their slanted tail in the fourth section entitled "**The BPPT slanted tail UAV**". Finally, the article remarks are concluded in the last section of this article.

UAV TAIL CONFIGURATION MODIFICATION: EXPLORING THE REQUIREMENTS OF THE CURRENT AERIAL MISSION

UAV has developed rapidly for various aerial missions. The mission expanded from a relatively simple task like the close range aerial photography and agricultural (Freeman & Freeland, 2015) to the more advanced mission like air quality measurements (Villa et al., 2016), weather monitoring (Samantaray & Narayan, 2014) and Search and Rescue (SAR) purposes (Sargent, 2017). Such advanced purposes required specific abilities in the paradox of the aircraft constraints. For example, the surveillance mission required the UAV to perform long-range flight in its limited fuel carried onboard. This type of contradiction has a way out if the aircraft can be configured to reduce its drag. These demanded missions required unconventional solutions provided by UAVs, thus, the conventional paradigm of aircraft design is not adequate anymore.

One of the conventional configuration is the T-tail construction for aircraft empennage. The separation of the vertical fin and the horizontal tailplane are intended to provide the uncoupled control of the longitudinal from the directional mode. Since the trailing edge of the horizontal tail is the elevator's location, thus the pitching movement due to the elevator deflection will not impel the yaw movement in the aircraft's directional mode. However, the attachment of the centerline axis of the horizontal tailplane on the top of vertical fin increasing the risk of flutter that might be triggered at that attachment. This T-tail flutter is a dynamic structural instability caused by the excessive vibration excited by the aeroelastic coupling between the aircraft's horizontal tailplane and its vertical fin (Murua et al., 2014).

In the contrary, the V-tail and inverted V-tail configuration combined the horizontal and vertical tail plane into a symmetrical pair of the slanted tailplane. The inclination arranged so both planes attached more firm to the aircraft fuselage and each of the plane's flexibility will not interfere each other. Since a single vertical fin created the flow interaction with the fuselage and contributed to the directional stability (Wen et al., 2013), then the construction of the slanted plane tail will double the vertical plane and increase the directional stability in flight. The dihedral angle of the slanted tailplane yields 5% improvement of the directional stability compared with the conventional tail (Musa et al., 2015). Such unconventional tail configurations are more suitable for UAVs since the absence of human onboard allowed more agility of flight for carrying out its mission.

Continuing the example previously explained, there are several trade-offs to reduce the UAV drag. One can scale down the wing dimension to reduce the drag, but unfortunately, narrowing the wing area will surely decrease the UAV lift. Similar trade-off also applied in the alternative of shrinking the main body compartment size. Such compression will reduce the drag too, but it also decreases the required volume of the payload carrier. These two trade-offs are likely limiting the UAV sensory capability since they will restrict the size of the onboard sensor, which also restricts its sensor weight capacity and surely degrades its sensing capabilities.

The acceptable trade-off for drag reduction resulted in the narrowing the tail area since it won't influence the payload carrying capability. By applying the slanted tailplane, especially the V-tail, the UAV drag can be reduced by 7 to 7.6% compared to the conventional tail (Vatandaş & Anteplıoğlu, 2015). However, narrowing the tail area might draw an impact to the aircraft controllability. Nevertheless, since UAV carries no person, it will be allowed to fly the maneuver beyond the human comfort threshold as long as the loads in-flight are bearable within the aircraft structures. Thus, the maximum G-force loading determined from the aircraft structural integrity, not from the pilot limitations (Reinhardt et al., 1999).

Thus, the ideas of modifying the tail configuration are applicable for UAVs. The goal of the tail modification is narrowing down the wetted area for the drag reduction. The modification is also intended to impose the desirable flight characters in accommodating the special purpose such as an agility of maneuver or to assist more stability in yawing. Figure 2 showed the result of modified tail designed UAV developed by the BPPT.

Figure 2. “Gagak” the Unconventional Tail UAV developed by the BPPT.

THE UAV UNCONVENTIONAL TAIL CONFIGURATION

The tail modifications resulted from the trade-offs in finding the acceptable solution for drag reduction of the aircraft. This drag reduction is important for the UAV strategic performances such as the longer time of endurance, the further distance of flight range, the less of fuel consumption and the shorter distance for taking off. Another objective of the tail modification is to achieve certain flight characteristics in supporting the UAV special missions.

The modifications are applied to reduce the UAV drag without impairing its stability and controllability. To obtain the drag reduction, the modifications are achieved by narrowing down the tail’s total wetted area. On the other hand, the modification had to conserve the adequate vertical projected area for directional stability purpose. Similarly, it should provide the sufficient horizontal projected area for longitudinal stability.

The trade-off leads to the application of the slanted tailplane, which joining the tailplanes into the V-Tail shape or the Inverted V-Tail shape. These slanted tail planes reduced the wetted area compared with the conventional tail, i.e. the sum of the horizontal and vertical tail plane that perpendicular each other. These slanted tail planes conserved the effective tail area in the vertical and horizontal projection.

The V-Tail Configurations

The idea of the V-tail aircraft had been implemented in a manned aircraft named the Beech V-Tail Bonanza. Since it was flyable in the manned aircraft concept, therefore the V-Tail will be applicable also for the UAV purposes. This tail configuration joins the horizontal and vertical plane into the V-shaped plane with the vertex attached at the aircraft rear. It utilizes the two slanted tail surface are fixed to serve as both horizontal and vertical stabilizers (FAA, 2016). The configuration is previously known as the “Butterfly Tail” for the similarity between the V-tail plane with the butterfly wings.

Located at the trailing edge of the V-tail, there is a deflectable segment called ruddervator for each of the tailplane. These ruddervators are mixing the role of the aircraft rudder and elevator. They behaved as an elevator when moving in the similar direction and with the equal deflection angles. They also performed as a rudder when moving in different directions. Unavoidably, these mixed roles required more complex control mechanism (FAA, 2016) in directing the aircraft movement.

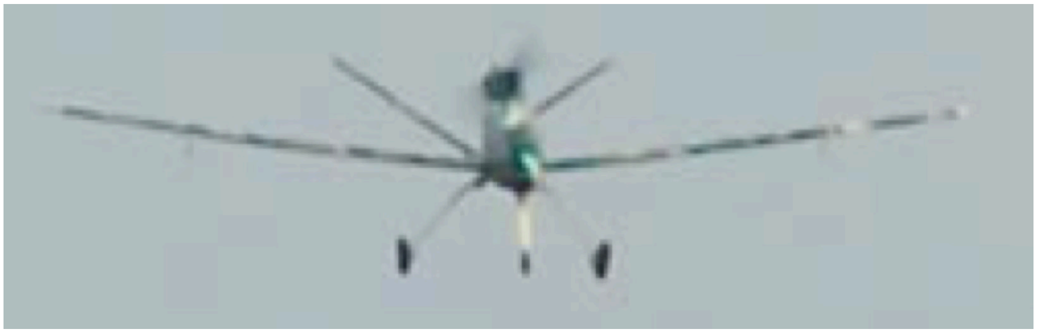


Figure 3. The BPPT's Flying V-Tail UAV.

The construction of conventional tail consists a simple attachment of each root of the tailplane to a fixed joint at the aircraft main body (Naidu et al., 2017). Instead of joining each plane, the V-tail configuration applied a more complex attachment that joined both of the planes with the aircraft rear, as seen in Figure 2 and Figure 3. This attachment of the tailplane vertex with the aircraft main body required special reinforcement to prevent the changes of its dihedral angle due to the bending load occurred in flight. Regardless of the complexity in the structural reinforcement for the vertex, the V-Tail excites more agility to the aircraft flight characteristics. The agility arises from the adverse roll-yaw coupling (Raymer, 1999) which appeared when the ruddervator deflected for yawing.

The unique design of the V-tail can also introduce a lightly damped Dutch roll mode (Rao, 1995) since it is more susceptible than the conventional tail (FAA, 2016). The Dutch roll interaction with the nonlinear characteristics of the V-tail might excite the dynamic stall. Following the absence of human onboard, then it is preferred to build the V-tail aircraft in the form of UAVs. Figure 3 showed the flying V-tail UAV developed by BPPT.

The Inverted V-Tail Configurations

Recently, the inverted V-tail UAVs appeared in the popular UAVs from Aerosonde (Cappello et al., 2016) to Predator (Shakarian et al., 2013). These UAVs are recognized by their long-range capabilities. They merged the horizontal plane and vertical fin into a common pair of the inverted V-shaped construction. Both of the tailplane installed with the higher location of the vertex and applied the downward inclination angles or the anhedral angle.

Figure 4. The Flying BPPT's Inverted V-tail UAV.

In the trailing edge of the inverted V-tail, there is a deflectable segment called the ruddervator at each plane. These ruddervators mixed the function the aircraft rudder and elevator. They served as the elevator when moving in the similar direction and with the equal angular magnitude. They act as the rudder when moving in different angles or directions. Similar to the V-tail construction, these mixed roles are also unavoidable and requiring more complex control mechanism. However, the electronic autopilot feature will be able to overcome this complexity.

The V-tail increased the yawing stability of the aircraft maneuvers. This stability increment arose from the proverse roll-yaw coupling (Raymer, 1999) when the ruddervator deflected to yaw the aircraft. The inverted V-tail compensate the tail area narrowing with this slight increment of the roll-yaw stability. This preferable characteristic of the inverted V-tail plane becomes the reason of its wide application. Figure 4 captured the flying moment of the inverted V-tail UAV developed by the BPPT.

THE BPPT SLANTED TAILPLANE UAV

The BPPT, stands for the *Badan Pengkajian dan Penerapan Teknologi* (translated as the Indonesian Agency for the Assessment and Application of Technology (http://www.bppt.go.id/)), is an Indonesian governmental institution which supported the development of unmanned technologies. Thus, to inseminate the advanced technology of unmanned systems for the nation's welfare purposes, BPPT has developed UAVs for Indonesia since 1999. Several missions to be assessed in the development including (Jenie, 2007):

- Observation of high-voltage electricity transmission line
- Cloud sniffer in weather modification
- Airborne data link system for tsunami early warning systems
- Hot spots identification in forestry fire
- Low-altitude hyper spectral for agricultural purpose

The design of these UAV followed the common emphasis on affordability, practicality and recoverability (Reinhardt et al., 1999). Moreover, to explore the state of the arts for the surveillance ability, the autonomous flight technology, special purpose camera, and advanced missions (Jenie, 2007), BPPT developed three types of large UAVs with 120 kg of Maximum Take-Off Weight (MTOW). The first type is the conventional T-tail BPPT-01A "Wulung", the second is the inverted V-tail BPPT-02A "Pelatuk", and the third is the butterfly tail, BPPT-01B "Gagak". In this article, the latter two equipped with the slanted tail will be assessed and analyzed.

Figure 5. Installation of the Tail Planes for the BPPT-02A "Pelatuk".

The BPPT's Unconventional Tail UAV

The earliest types of Indonesian large UAVs which equipped with the slanted tailplane were developed by the BPPT. *Gagak*, (transl.: the crow), was one of Indonesian UAV type with the V-tail empennage. The other type was *Pelatuk*, (transl.: the woodpecker), one of the Indonesian earliest UAV with the inverted V-tail empennage. Both are the BPPT's UAVs that capable of carrying 120 kg of MTOW included 20 kg of payload.

The slanted plane tail, include the V-tail and inverted V-tail provided many advantages over the conventional T-tail configuration. Such advantages are including the hind packaging and the portable installation concept that UAV can be assembled and deployed by two-person (Babu et al., 2016). Figure 5 confirming such advantages in the BPPT-02A "Pelatuk" UAV. Since the tail boom located high enough from the ground, from here the two people are able in assembling the tailplane in their normal standing or squatting position as necessary.

The slanted plane empennage combined the conventional horizontal plane and vertical fin into a pair of the slanted plane. The combination reduced the aircraft drag since the slanted tail has narrower wetted area compared with the sum of its projected area in the vertical and horizontal plane (Raymer, 1999). The reduction of drag also occurred since the planes combination lessen the number of corners, from two in the T-tail into one corner in the slanted tail, for such corner produced the additional interference drag at the empennage. The drag in the conventional tail was reported to be 7 to 7.6% higher than the slanted tail UAV (Vatandaş & Anteplıoğlu, 2015).

The BPPT-01B "Gagak": The V-Tail Indonesian UAV

Configured with the Butterfly Tail and low wing placement, the BPPT-01B "Gagak" was designed to perform the Lo-Hi-Lo mission, i.e. taking-off and cruising at **low** altitude, then climb to **high** altitude to perform the mission, finally descent to **low** altitude on its way back to the base. The intended mission required a UAV with higher agility in its flight characteristics. The agility can be achieved using the low placement of wing and using the adverse roll-yaw effect to support its maneuver. The detail explanations are presented in the following paragraphs.

Figure 6. Engine Maintenance of BPPT-01B "Gagak".

The V-tail configuration avoids the contact between the empennage to the ground and providing the free space for further modification of landing parachute opening (Kuzdas & Pátek, 2015). The V-tail empennage also introduced the adverse roll-yaw moment (Raymer, 1999) when the pilot applying the ruddervator deflection. The combination between the low wing placement that induced the inverted pendulum effect to the aircraft roll stability and the V-tail is increasing the flight agility (Jenie, 2007) of the BPPT-01B "Gagak". These advantages provide the support for the maneuvering mission of patrolling and monitoring. Since the Indonesia is covered by wide areas of ocean, the maritime monitoring or observation mission will be suitable to be accomplished by large UAV (Zheng et al., 2017) including the *Gagak*.

The engine directly rotates the propeller from its main axis without additional gear or belt or shaft to avoid more losses due to friction. The arrangement also distanced the propeller from the V-tail. This proper distance of propeller from the V-tail maintains the maximum achievable range and endurance of the UAV. Otherwise, for the propeller installed close to the empennage will reduce the lift to drag ratio (Chen et al., 2015) which decrease the achievable range and endurance of the UAV. The combination of low wing and low tail boom configuration pushed the engine placement on the top of the fuselage's rear. Thus, as shown in Figure 6, the engine is easier for installation or uninstallation process and for maintenance activities. In the operation and maintenance activities, the UAV must meet a condition to allow a simple maintenance process in a short flight, allowing a quick preparation, able to maneuver easily in the designed flight and ensure the safety of maintenance crew (Prisacariu et al., 2014). Several maintenance items including the propeller check and propeller changing to meet the desired flight performance for a specific mission. As a rule of thumb, the higher pitch propeller is suitable for shortening the take-off distance but it gave a smaller rate of climb. Conversely, the lower pitch propeller supports the higher rate of climb and lower fuel rate of cruise flight, but it required longer take-off distance for airborne. The photograph in Figure 6 showed the easiness for propeller change in normal standing activity.

Although the BPPT-01B "Gagak" had higher agility in flight, its aerodynamic aspects still showing good results, as it confirmed in the wind tunnel test (Wijiatmoko & Daryanto, 2016). The high value of lift coefficient of the BPPT-01B "Gagak" correlated with the lower effective fuel consumption rate. This correlation occurred since the higher value of lift coefficient lessen the airspeed needed to carry the vehicle's weight in flight. The speed reduction also lowered the fuel rate in flight. The fuel rate reduction then contributing to the better flight endurance and an adequate flight range.

The BPPT-02A "Pelatuk": The Inverted V-Tail Indonesian UAV

With its inverted V-tail and high wing configuration, the BPPT-02A "Pelatuk" was intended to perform the Hi-Lo-Hi mission. The Hi-Lo-Hi means that the UAV started with taking-off, climb and cruising in **high** altitude, after that descending to the **low** altitude to perform the mission, then climb again to **high** altitude on its way back to the base. The intended mission required a UAV with a short take-off and landing (STOL) feature and a higher degree of flight stability. These features will be achieved by placing the engine in the lower part fuselage and applying the inverted V-tail configuration.

Figure 7. Low Engine Placement of BPPT-02A "Pelatuk".

Contrary to the previous V-tail, the inverted V-tail empennage provides the proverse roll-yaw moment (Raymer, 1999) when the pilot applying the ruddervator deflection. This proverse moment increases the flight stabilization when BPPT-02A "Pelatuk" performing the yaw movement. The benefit from a better flight stabilization supports the narrow area photography or a close-range surveillance. Consequently, the BPPT-02A "Pelatuk" was initially proposed for the forestry and maritime patrol (Jenie, 2007).

The high placement of the wing decreases the possibilities of damage if the landing is not smooth or landing at a rough terrain (Kuzdas & Pátek, 2015). The high boom construction intended to provide a sufficient flexibility for the tilt angle of the engine's thrust. As if the engine tilted downward, it provided the Short Take-Off Landing (STOL) feature for the BPPT-02A "Pelatuk" (Jenie, 2007). Thus, the UAV can take off from remote islands with short runway available. One of the predecessors of the STOL featured UAV is the Pioneer I RPV (Reinhardt et al., 1999) which has a smaller dimension, hence it can be launched from a ship.

Figure 7 showed the engine location of BPPT-02A "Pelatuk" at the lower rear fuselage. The engine placement below the tail boom, as seen in Figure 7, allowed a pitching rotation of the engine in a lateral axis. The STOL feature occurred when the engine tilted downward so the vertical projection of the engine thrust provided extra lift. The sum of the wing provided lift and the additional lift from that tilted engine thrust reduced the UAV's lift-off speed and shortening the take-off distance.

CONCLUDING REMARKS

This article has addressed the important remarks of the slanted tailplane of Indonesian UAV developed by the BPPT. Besides having the drag reduced by the implementation of the slanted tailplane, the BPPT-01B "Gagak" also gain more agility in flight from the V-tail configuration while the BPPT-02A Pelatuk obtained more stability with the inverted V-tail configuration. The advantages also arise with the disadvantages occurred such as more complex of the surface control mechanism, and the nonlinear characteristics due to the coupling between rolling and yawing. However, the priority in achieving the mission accomplishment circumvent those disadvantages. Furthermore, the unmanned concept—since there's no person onboard—is relaxing the design constraint especially the ones related to the passenger comforts.

Finally, the advantages of the slanted tail have increased the number of UAV equipped with that tail configuration. Innovative solutions appear in all stages of making unmanned air product from the conceptual, design, prototyping and manufacturing resources (Prisacariu et al., 2014). These slanted tail configurations have marked the innovation in the aircraft design. Hence, these advantages are able to motivate further developments for *Gagak* and *Pelatuk*, the BPPT multipurpose Indonesian UAV, to perform the strategic missions for the welfare of the nation, especially in this middle era of the drone race (Boyle, 2015).

REFERENCES

Babu, P. H., S.A, R. A., Basha, S. M., & P, S. (2016). Design and Analysis of V-Tail Unmanned Air Vehicle (UAV) for Surveillance Application. International Journal of Engineering Trends and Technology (IJETT), 32(3), 119-123. doi: 10.14445/22315381/IJETT-V32P221

Boyle, M. J. (2015). The Race for Drones. Orbis, 59(1), 76-94. doi: https://doi.org/10.1016/j.orbis.2014.11.007

Cappello, F., Ramasamy, S., & Sabatini, R. (2016). A low-cost and high performance navigation system for small RPAS applications. Aerospace Science and Technology, 58 (Supplement C), 529-545. doi: https://doi.org/10.1016/j.ast.2016.09.002

Cavallaro, R., & Demasi, L. (2016). Challenges, Ideas, and Innovations of Joined-Wing Configurations: A Concept from the Past, an Opportunity for the Future. Progress in Aerospace Sciences, 87, 1-93. doi: https://doi.org/10.1016/j.paerosci.2016.07.002

Chambers, J. R. (2005). Innovation in flight: research of the NASA Langley Research Center on revolutionary advanced concepts for aeronautics (Vol. 39): NASA History Division.

Chen, G., Chen, B., Li, P., Bai, P., & Ji, C. (2015). Numerical Simulation Study on Propeller Slipstream Interference of High Altitude Long Endurance Unmanned Air Vehicle. Procedia Engineering, 99 (Supplement C), 361-367. doi: https://doi.org/10.1016/j.proeng.2014.12.548

Collins, R. L. (2012). What Was Wrong With V-Tail Bonanza Pilots? Air Facts: the Journal for Personal Air Travel--by pilots, for pilots. Retrieved from Air Facts: the Journal for Personal Air Travel-by Pilots, for Pilots website: http://airfactsjournal.com/2012/06/tail-tale-what-was-wrong-with-v-tail-bonanza-pilots/

FAA. (2016). Flight Control Pilot's handbook of aeronautical knowledge. Oklahoma City: Airman Testing Standards Branch, FAA, US Department of Transportation.

Filippone, A. (2000). Data and performances of selected aircraft and rotorcraft. Progress in Aerospace Sciences, 36(8), 629-654. doi: https://doi.org/10.1016/S0376-0421(00)00011-7

Freeman, P. K., & Freeland, R. S. (2015). Agricultural UAVs in the U.S.: potential, policy, and hype. Remote Sensing Applications: Society and Environment, 2, 35-43. doi: https://doi.org/10.1016/j.rsase.2015.10.002

Grissom, A. R., Lee, C., & Mueller, K. P. (2016). Innovation in the United States Air Force: Evidence from Six Cases: RAND Corporation-Project Air Force Santa Monica United States.

Jenie, S. D. (2007). The UAV Research, Development and Engineering Technology (In Indonesian). Paper presented at the National Seminar for UAV Technology, Jakarta.

Kroo, I. (2004). Innovations in Aeronautics The 42nd AIAA Aerospace Sciences Meeting and Exhibit: American Institute of Aeronautics and Astronautics.

Kuzdas, J., & Pátek, Z. (2015). SOKOL Unmanned Aerial System. Advances in Military Technology, 10(1), 99-109.

Mali, S., Muthe, Y., & Kolhe, V. (2016). Concorde-2: An Overview. International Advanced Research Journal in Science, Engineering and Technology, 3(1), 170-173. doi: 10.17148/IARJSET/ICAME.32

Murua, J., Martínez, P., Climent, H., van Zyl, L., & Palacios, R. (2014). T-tail flutter: Potential-flow modelling, experimental validation and flight tests. Progress in Aerospace Sciences, 71, 54-84. doi: http://dx.doi.org/10.1016/j.paerosci.2014.07.002

Musa, N. A., Mansor, S., Ali, A., & Che Man, M. H. (2015). Effect of Tail Dihedral Angle on Lateral Directional Stability due to Sideslip Angles 53rd AIAA Aerospace Sciences Meeting: American Institute of Aeronautics and Astronautics.

Naidu, V. N., Babu, P. H., & Kesava, T. G. (2017). Design and Analysis of V-Tail Unmanned Air Vehicle (UAV) for Surveillance Application. International Journal of Core Engineering & Management, Special Issue (NCETME-2017), 135-142.

Prisacariu, V., Boscoianu, M., & Luchian, A. (2014). Innovative solutions and UAS limits. Review of the Air Force Academy, 26(2), 51.

Rao, K. D. (1995). Modeling nonlinear features of V tail aircraft using MNN. IEEE Transactions on Aerospace and Electronic Systems, 31(2), 841-845. doi: 10.1109/7.381934

Raymer, D. P. (1999). Aircraft design: A conceptual approach: American Institute of Aeronautics and Astronautics.

Reinhardt, J. R., James, J. E., & Flanagan, E. M. (1999). Future employment of UAVs: issues of jointness. Washington DC: National Defense University.

Richardson, T. S., McFarlane, C., Isikveren, A., Badcock, K., & Da Ronch, A. (2011). Analysis of conventional and asymmetric aircraft configurations using CEASIOM. Progress in Aerospace Sciences, 47(8), 647-659. doi: https://doi.org/10.1016/j.paerosci.2011.08.008

Rizzi, A. (2011). Modeling and simulating aircraft stability and control—The SimSAC project. Progress in Aerospace Sciences, 47(8), 573-588. doi: https://doi.org/10.1016/j.paerosci.2011.08.004

Samantaray, S., & Narayan, V. (2014). Weather Monitoring Unmanned Aerial Vehicle. Advanced Research in Electrical and Electronic Engineering, 1(2), 69-72.

Sargent, A. (Producer). (2017, March 20). Drone tested by Albany sea rescuers at Salmon Holes set to help in emergencies. ABC News. Retrieved from http://www.abc.net.au/news/2017-03-19/drone-to-be-used-for-albany-sea-rescues-after-salmon-holes-test/8367318

Shakarian, P., Shakarian, J., & Ruef, A. (2013). Chapter 10 - How Iraqi Insurgents Watched U.S. Predator Video—Information Theft on the Tactical Battlefield Introduction to Cyber-Warfare (pp. 185-196). Boston: Syngress.

Smith, N. (2016). Concorde: End of the dream? Engineering & Technology, 11(1), 60-64. http://digital-library.theiet.org/content/journals/10.1049/et.2016.0128

Sun, Y., & Smith, H. (2017). Review and prospect of supersonic business jet design. Progress in Aerospace Sciences, 90, 12-38. doi: https://doi.org/10.1016/j.paerosci.2016.12.003

Vatandaş, O. E., & Anteplıoğlu, A. (2015, 16-19 June 2015). Aerodynamic performance comparison of V-tail and conventional tail for an unmanned vehicle. Paper presented at the 2015 7th International Conference on Recent Advances in Space Technologies (RAST).

Villa, T., Gonzalez, F., Miljievic, B., Ristovski, Z., & Morawska, L. (2016). An Overview of Small Unmanned Aerial Vehicles for Air Quality Measurements: Present Applications and Future Prospectives. Sensors, 16(7), 1072.

Wen, J., Deng, X. Y., Wang, Y. K., & Ou, S. (2013). Flow Investigation on the Directional Instability of Aircraft with the Single Vertical Tail. Procedia Engineering, 67, 328-337. doi: http://dx.doi.org/10.1016/j.proeng.2013.12.032

Wijiatmoko, G., & Daryanto, Y. (2016). The Aerodynamic Test of Unmanned Aerial Vehicle Model with V-Tail Empennage (in Indonesian). 9. Retrieved from www.researchgate.net website: https://www.researchgate.net/profile/Yanto_Daryanto/publication/311409276_Pengujian_Aerodinamika_Model_Uji_Pesawat_Udara_Nir_Awak_dengan_Empennage_berjenis_V-Tail/links/5844c76b08ae2d217566d4df.pdf

Zheng, Q., Klemas, V. V., & Yan, X. H. (2017). Progress in Ocean Remote Sensing: Overview Reference Module in Earth Systems and Environmental Sciences: Elsevier.

JOURNAL OF UNMANNED AERIAL SYSTEMS

WWW.UASJOURNAL.ORG