UAV POSITIONING DATA DETERMINED VIA ARUCO TAGS FOR AIRCRAFT SURFACE INSPECTION

THESIS

Caleb B. Schmidt, Captain, USSF

AFIT-ENG-MS-22-M-059/AFIT-ENV-MS-22-M-258

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENG-MS-22-M-059/AFIT-ENV-MS-22-M-258

UAV POSITIONING DATA DETERMINED VIA ARUCO TAGS FOR AIRCRAFT SURFACE INSPECTION

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Electrical Engineering

and Master of Science in Systems Engineering

Caleb B. Schmidt, B.S.C.E.

Captain, USSF

March 24, 2021

AFIT-ENG-MS-22-M-059/AFIT-ENV-MS-22-M-258

UAV POSITIONING DATA DETERMINED VIA ARUCO TAGS FOR AIRCRAFT SURFACE INSPECTION

THESIS

Caleb B. Schmidt, B.S.C.E.
Captain, USSF

Committee Membership:

Robert C. Leishman, Ph.D
Chair

Clark N. Taylor, Ph.D
Member

David R. Jacques, Ph.D
Member

AFIT-ENG-MS-22-M-059/AFIT-ENV-MS-22-M-258

Abstract

Aircraft are frequently inspected to ensure that military and civilian safety standards are adhered to. These inspections are performed pre- and post-flight and are currently performed by trained maintenance personnel. These inspections take multiple hours per aircraft per day and the goal of this research is to further develop a vision-based inspection system utilizing an Unmanned Aerial Vehicle (UAV). This work furthers the automation of aircraft surface inspection by using ArUco tags to determine the position of the UAV during aerial inspections. The ArUco tag-based position data was then compared to a highly accurate infrared motion capture system to determine the viability of this approach for accurate positioning of the vehicle. This information will be utilized in future works to determine the location of surface flaws or damage to alert maintainers. This work includes flight experiments with two different UAVs to perform a system viability comparison in order to determine the optimal solution for a fully automated aircraft surface inspection system.

Table of Contents

Page

List of Figures

Figure Page

Figure Page

List of Tables

UAV POSITIONING DATA DETERMINED VIA ARUCO TAGS FOR AIRCRAFT SURFACE INSPECTION

I. Introduction

1.1 Research Motivation

This research pursues ArUco tag-based positioning of a Unmanned Aerial System (UAS) during aircraft surface inspection. The overall intent of this research is to reduce total man-hours required for inspection while increasing inspection effectiveness and safety. This objective can be realized by replacing maintainers with a UAS completing the aircraft surface inspection. The UAS system would utilize vision-based navigation to fly within a GPS-denied space while detecting and documenting the locations of aircraft surface flaws to repair. In addition to the general use case of the Department of Defense (DoD) for routine aircraft maintenance and post-flight maintenance, this system targeted a use-case of inspecting stealth coated aircraft surfaces. Stealth aircraft require heightened levels of surface maintenance due to their reliance on coating materials. Maintainers work within the standards of human vision and perform their inspections from a distance of no further than three feet from the aircraft. This distance can be difficult and dangerous to achieve with large aircraft. This autonomous system would remove the need for the physical component of the inspection for maintainers, while performing the inspection at a much faster rate than possible with current practices. The usage of an automated flaw detection system would additionally aid the United States Air Force (USAF) by increasing the total flaws found while decreasing total inspection times. [2] [3] [4] [5] [6]

1.2 Research Objectives

The primary goal of this research is to develop a process utilizing an Unmanned Aerial Vehicle (UAV) to visually inspect military aircraft to detect skin surface flaws and is an expansion on the research performed by previous Air Force Institute of Technology (AFIT) students. Prior students focused on optimal pathing of the inspection flight patterns and achieving rapid flight times [7]. Prior research used a different UAV to inspect a 1/7 scale F-15 model within the AFRL Vicon chamber. Differing from previous research, this work performs a comparison between a COTS aircraft and an AFIT built aircraft. Both vehicles were flown within the AFRL Vicon chamber as well as the United States Air Force Museum (USAFM) Restoration Hangar. The AFIT ANT Center vehicle was modified to accomplish this specific research while creating a method to document the position of the UAV relative to the inspected aircraft for future use in a flaw detection algorithm created by another student. To achieve this goal and aid in the eventual transition to active use by the USAF there are a series of sub-goals that need to be achieved. This work entails a comparative element between two different potential capabilities.

The first is a Commercial off-the-shelf (COTS) capability, through the Skydio 2. This UAV is equipped with vision-based tracking, node-based navigation, and collision avoidance out of the box that are pertinent to this use case. The second UAV being investigated is an ANT Center custom-built hexrotor. This hexrotor has a 4K camera and a control system that allows for user changes. Both systems have been described in individual Systems Architectures built to contain all the components and connections between them. This has been done as a tool to guide the sponsors with choosing a system that fulfills their goal of safe and rapid autonomous aircraft inspection. This work has a focused analytic section based on the position accuracy of the two tracking systems that will be used. The flights performed by these UAVs were conducted within the Air Force Research Laboratory (AFRL) Vicon chamber. Motion capture systems, such as the the IR-based camera tracking system from Vicon, are well-regarded as a source of positional truth data. During this research both UAVs were flown inside the Vicon chamber to record truth data relative to the scale model aircraft in the chamber. As a full motion capture system would be time and cost prohibitive in a real-world hangar environment. This research utilized ArUco markers to create a second source of position data. The position data of the two locating methods was then compared to determine differences in accuracy. This was done to ensure that in a real-world hangar scenario the location truth data needed for identifying the locations of aircraft skin flaws from UAV imagery would be obtainable. The overarching goal of this research is the creation of a UAV system to document the position of flaws on an aircraft in real-time while displaying a list of flaw locations to a maintainer within the aircraft hanger.

1.3 Research Contributions

This research contributed the following to the area of UAV based inspection:

1. **Viability Study Comparing UAVs**: The sponsor of this research needs a field-able system that can perform the inspection as desired. Using a COTS aircraft and an AFIT created aircraft explored the benefits and downfalls of each type of system and will help the sponsor be more informed about which type of system they will want to employ for their inspections.

2. **System Architectures**: This work created system architectures of each UAV utilized to better inform the sponsor of the hardware needs for the creation of an eventual end product.

3. **Pose Estimation from ArUco Tags**: This work modified open-source code

from Open-CV libraries to create a method of finding the position of a UAV performing an aircraft inspection relative to pre-positioned ArUco tags surrounding the inspected aircraft. This work pursued two different methods for pose estimation focusing around ArUco OpenCV functions as well as PNP OpenCV functions. This information will be utilized by another student to determine the location of the aircraft skin flaws recorded in the inspection videos.

1.4 Document Overview

This thesis is laid out across five chapters. Chapter II is focused around prior methods of aircraft inspection leading to this work's vision-based inspection technique as well as information behind the ArUco tags that were used for position data during inspection. Chapter III further explores the usage of ArUco tags within this research and how the flight tests were performed and the inspection video was used. Chapter IV covers the results from the comparison of position data from the ArUco positioning method with the position data from the Vicon chamber system. Chapter V focuses on the conclusions drawn from the results of this research and the recommendations for the sponsor of this project and future researchers around this topic area.

II. Background and Literature Review

This chapter discusses alternative methods for aircraft inspection, the current state of UAV-based aircraft inspection, background in using vision-based positioning techniques, and information about the utilization of systems architecture for effective prototyping and documentation. Section 2.1 discusses previous works surrounding the field of Aircraft Inspection via UAV. Section 2.2 focuses on the background of general aircraft inspection techniques as well as different options that could be considered for inspection. Section 2.3 discusses the use of OpenCV, ArUco tags and other options for fiducial marker detection. Section 2.4 provides a general overview of systems architecture and the purpose of its inclusion in this work.

2.1 Previous Works

2.1.1 ANT Center

Previous work done by the Air Force Institute of Technology (AFIT) Autonomy and Navigation Technology (ANT) Center was centered around path planning for a UAV performing an aircraft surface inspection; the work focused on utilizing coverage path planning algorithms in simulated and experimental flight testing environments[7]. Captain Silberberg's research demonstrated that these path planning algorithms were effective at creating routes for a UAV to fly over an aircraft's surface. The experimental flights performed in his work were also conducted within the Air Force Research Laboratory (AFRL) Vicon chamber over the 1/7 scale F-15. These experimental flights laid the groundwork for utilizing Vicon motion-capture data for position information of the UAV, and the trials in this work were modeled off of the flight paths he created.

2.1.2 Related Inspection Research

Additional related research pursuits that were utilized in the development of this work discussed UAV trends in civil aviation and vision-based pose estimation through landmarking[8, 9]. Bugaj et al. discussed pre-flight inspection requirements and searched for a solution to replace the current use of human visual inspections[8]. Their research aimed to create a ”Smart Hangar” that utilizes UAVs with attached cameras to create 3D models of inspected aircraft. Flights performed with their UAV were done outside and around a twin-engined aircraft and were exclusively flown around the aircraft instead of over to avoid potential damage to the aircraft. The research performed in this paper was expanded on in this work by performing inspection flights within a hangar and performing flights above the vehicle being inspected.

Cazzato et al. aimed their research at detecting the aircraft pose using computer vision without fiducial markers [9]. This research utilized utilized natural landmarks to remove the requirement of fiducial markers to find the aircraft pose during inspection flights. To create these landmarks they used a single image of distinct areas of the aircraft surface, i.e. a photo of a flag painted on the tail or an image of the airline name, and created a bounding box on the corners of the landmark. They then used a computer vision algorithm to detect these pre-made landmarks in real-time during inspection flights to determine the aircraft pose. Instead of creating these aircraft specific landmarks, this work used ArUco tags to determine the UAV pose.

The research performed by Ariante et al. used a UAS platform with Light Detection and Ranging (LIDAR) and an Inertial Measurement Unit (IMU) to determine the position of the vehicle relative to detected surfaces[10]. This use case is very similar to the needs of the UAV Pose Estimation problem and contains ideas that were considered as options for this project. LIDARs have been used by the ANT Center in the past as they are very capable at determining distances from surfaces. Ariante et al. utilized the LIDAR on their aircraft to detect obstacles and determine the distance from the ground during flight. These distances to surfaces in the environment were used in combination with the velocity data recorded from the IMUs improve the system's ability to determine its position and attitude within its known environment. These capabilities were considered as options to be included on-board the Hexrotor aircraft utilized in this work and would be great additions in future works to improve the positioning determination of the aircraft.

A key area of the field of aircraft inspection focuses on the improvement of the precision and accuracy of the inspections being performed. Research has been performed to discover cracks, missing paint, dents, and even detection of screws on the fuselage of aircraft. Miranda et al. expanded their previous work in UAV-based aircraft inspection by focusing in on detecting all screws that should be visible on the fuselage of a given vehicle [11]. They did this by performing the aerial inspection, creating Zones of Interest (ZoI), and comparing what was detected within those zones to the known locations of the aircraft screws. This process of creating ZoI could prove useful in an expansion of the research performed in this work as it would aid the users or machine learning algorithms in detecting surface flaws using known imagery of aircraft in states of good repair.

The work performed by Jordan et al. regarded the use of UAVs for the inspection of power lines and other infrastructure required for power transmission [12]. Though the subjects being inspected were different, the fundamental need for precise detection and collision avoidance capabilities were shared with this work's needs. This research focused on discussing the electronic inspection techniques that have been used in their field in the past and whether or not they proved effective. The methods discussed for detection were all computer-vision based including the following sensors: Pan Tilt Zoom (PTZ), visible-light, thermal, and monocular cameras. Their paper discusses thoroughly a large amount of sensors that have been used in inspection capacities on UAVs for a large variety of subjects including power lines, buildings, and bridges. This large array of sensors is a useful starting point for any researcher that wants to perform UAV computer-vision inspections and wants help with sensor selection. This research and past projects by the ANT Center determined the selection of the Mako camera used on the Hexrotor in this work.

2.2 Aircraft Inspection Techniques

Pilots perform pre-flight visual inspections before every flight commercially, privately, and in the military. These pre-flight assessments are performed to ensure that airworthiness standards are met and the plane appears to be mechanically safe to fly [13, 8]. General Visual Inspection (GVI)s are also performed by aircraft maintenance personnel. GVIs are performed at least every two days for commercial airlines and after every flight for military aircraft [14]. While all aircraft need to be regularly inspected for potential maintenance concerns, any chips or scratches on the surface of a stealth aircraft can affect the Radar cross-section (RCS), which will make an aircraft easier for adversaries to discover. Current maintenance practices require many hours for a normal, post-flight inspection.

Beyond visual maintenance inspections, other inspection methods are typically performed utilizing Non-destructive testing (NDT) techniques [15]. NDT uses a variety of techniques to inspect aircraft during regular maintenance. These techniques are used to discover structural damage that cannot be discovered by simple observation of the airframe. Some of these methods include ultrasonic, x-ray, resonance testing, and infrared. These structural tests are often performed every 100 flight hours, and due to the cost of these systems and time to use them they are less economical choices. The combination of the price of some of these dedicated inspection systems as well as the scale of aircraft have lead many researchers to attempt inspection using automated systems including robots traversing the surface of aircraft using suction and UAVs with a variety of sensors [2, 3, 16].

2.2.1 Airborne Inspection

Aerial-based inspection was chosen by the sponsor of this work to realize the goal of aircraft surface inspection [7]. Due to costs of UAV technologies and sensors decreasing over time while sensor and capabilities and battery lifetimes have increased, UAV-based aircraft inspection is more feasible and effective than ever before. The applications for airborne inspections are numerous and research efforts have included bridge, building, road, and aircraft inspections [6, 17, 9].

Inspection techniques that were studied for guidance on this research project were focused on vision-based inspection. Other methods used for aircraft inspection studied were LIDAR, Radar, and Sonar techniques. During this background research it was determined that computer-vision techniques would be used with high frame rate, high resolution cameras on-board UAVs as they would be the most effective at scanning the surface of aircraft in the eventual use-case for this system. LIDAR could be used effectively in later experiments in tandem with the primary cameras on the UAVs to have more accuracy in determining the position of the scanning vehicle[18, 19, 20].

2.3 Marker Detection and OpenCV

Marker detection can be performed by computer vision systems. [21] These systems process fiducial markers within an image to detect the position and rotation of the marker in the frame. Some examples of these markers are Quick Response (QR) codes, ArUco tags, and AprilTag. These markers can be encoded with textual data that can point towards a website or, in the case of ArUco markers, binary squares tied to a specific value[22]. For this work, ArUco tags were utilized. These tags were selected as the libraries created around them are very popular and are heavily used in computer vision applications. The use cases for each set of fiducial markers has been determined by the way each style of tag has been used in the open-source computer vision community. QR codes are used in a very broad set of applications including links to menus, product information, and generally point to static sets of information. ArUco and AprilTag markers are used in similar capacities for real-time and post-processed pose estimation for computer vision and were looked at as better solutions for this research. Due to ArUco tags being more established with OpenCV and being used more often in general they were selected over AprilTags.

OpenCV was selected to perform the camera posing via computer vision techniques for this work. OpenCV was originally created by Intel and released in June 2000. This library was chosen due to the amount of computer vision tools that it contains and the large amount of prior work performed using it, as it is all open-source. To estimate the pose and the UAV position accurately the OpenCV functions surrounding ArUco tag detection, pose estimation, and Perspective N- points Problem (PnP) were utilized. These key functions used and the process of developing the software for this work is further laid out in detail in Chapter III.

2.3.1 Perspective N–points Problem

The problem of finding the position of the camera relative to points detected within the image frame recorded by the camera is described as the Perspective N-points Problem.[23] This problem utilizes a set of n 3D points in the world frame, in this research the Vicon frame, and the corresponding 2S projections in the image to estimate the pose of the camera creating the images. Figure 26 demonstrates PnP as sourced from the OpenCV documentation.

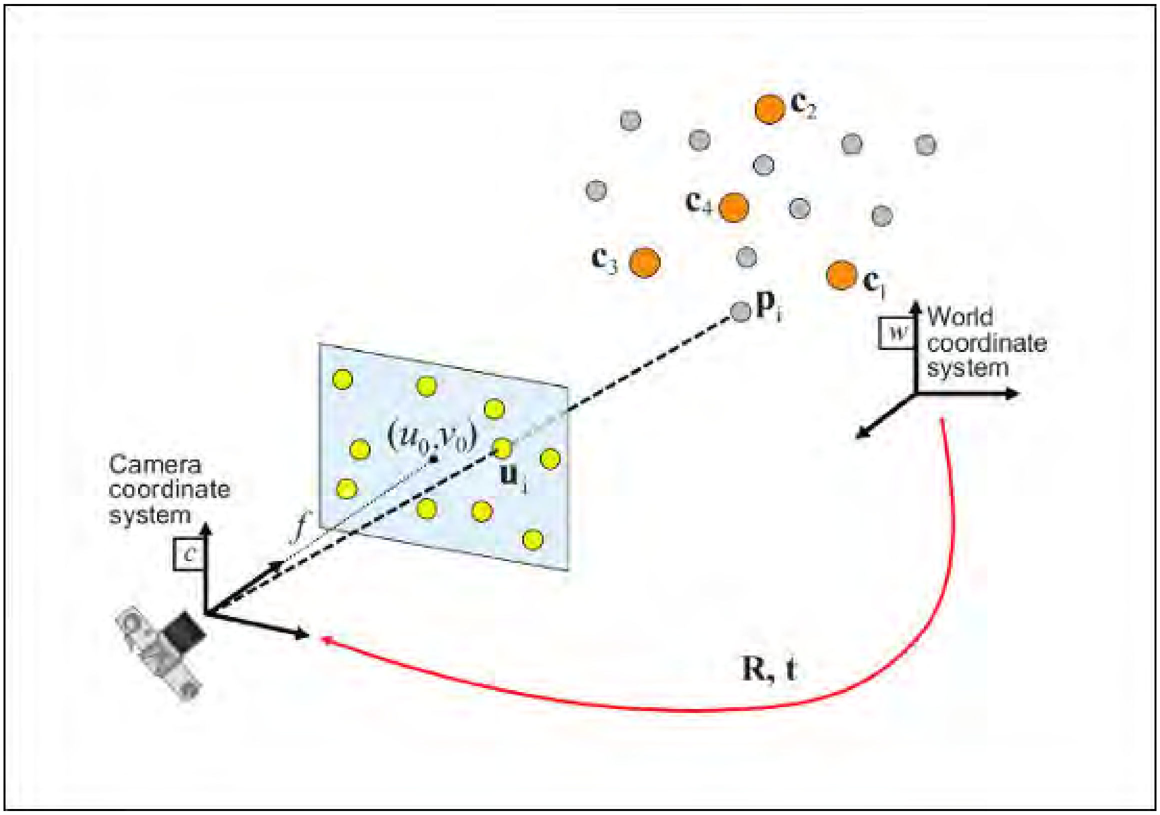

Figure 1: OpenCV: Perspective N-point Problem[1]

The diagram in Figure 26 was key in the utilization of the PNP functions contained within the OpenCV library. The illustration provided guidance to use the camera data and the Vicon data with the right camera orientation and world frame orientation. The use of these functions is further described within Chapter III.

2.4 Systems Architecture

Systems architectures are created to describe the inter-workings of a system's components. These architectures define the physical structure, the behavior and different perspectives or views of the system. [24] There are many methods of creating systems architectures and this work created the architectures used in this research utilizing

Model-Based Systems Engineering (MBSE) techniques. Cameo Systems Modeler, also referred to as Computer Aided Three-Dimensional Interactive Application (CATIA), is the MBSE environment that AFIT regularly uses and was also used in this work. System architectures were created for both UAVs utilized in this work. These architectures used the AFIT Small Unmanned Aerial System (SUAS) Reference Architecture as their foundation, and captures of the Physical Decomposition and Internal Block Diagram (IBD) of each system are included in Chapter III.

2.4.1 Advantages for Projects

This research project is far from the first time that MBSE techniques or system architecture techniques have been used to document military systems or UAVs [25, 26, 27]. Due to increased adoption of MBSE there have been discussions on the effectiveness of digitally modeling a system. Henderson et al. discuss the results of MBSE in literature in an attempt to determine whether or not there is value to this practice. Their results were inconclusive with the current amount of studies performed on model effectiveness. However, creating systems architectures for the systems involved with this work were found to be of value to the sponsor of the research as a means of documenting the components making up each vehicle as well as the connects to achieve the final system. Using these architectures will enable the sponsor to make decisions in the future on an eventual end-case system.

III. Methodology

This chapter describes the methodology used in the testing designed and executed to acquire the position data. This section also describes the hardware that each UAV utilizes, a systems architecture for each aircraft, and the software used to capture the data in flight. The overall layout of this chapter is as follows: Section 3.1 is a description of test articles flown during testing, the sensors on each aircraft, and the systems architecture for each UAV. Section 3.2 describes the process of determining the pixel size of flaws on a aircraft panel to aid in guiding future machine learning processes to detect flaws. Section 3.3 describes the dockerized software packages that were utilized to record flight footage and telemetry data in-flight. Section 3.4 describes each of the locations that were utilized during flight testing during this research. Section 3.5 describes the steps performed during the final experimental flights and objectives for each flight.

3.1 Flight Test Articles

This section covers each test article flown in this research's experimental flight in detail. Both systems were flown in identical flight conditions to enable this work to include a system viability comparison for future works.

3.1.1 Skydio 2

The Skydio 2 is a Commercial off-the-shelf (COTS) UAV that was selected due to the substantial amount of capabilities offered with a low price point. These capabilities are described in this subsection. Figure 2 displays the Skydio 2 at the origin point of the Vicon Chamber.

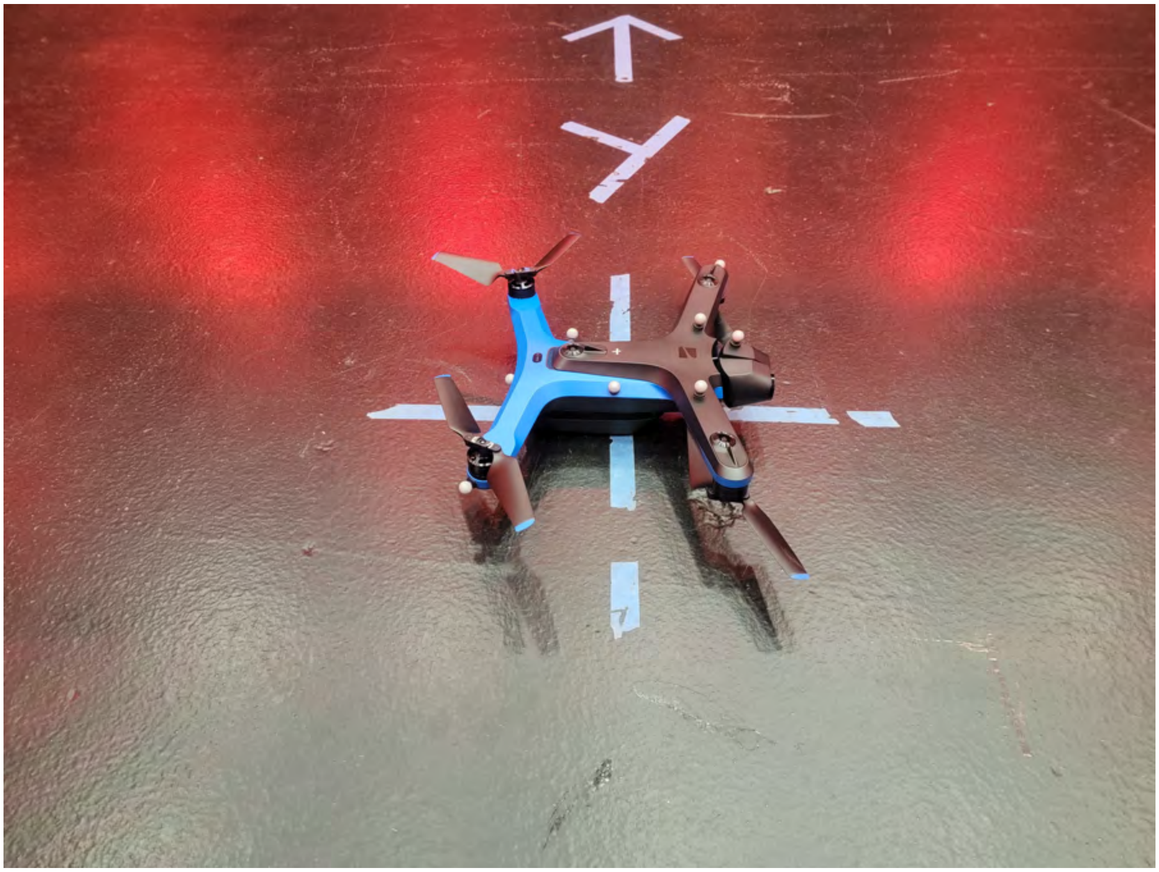

Figure 2: Skydio 2 pre-flight in the Vicon Chamber

The Skydio 2 is manufactured by Skydio located in Redwood City, California. The Skydio 2 is a four-armed quadrocopter measuring at 223 x 272 x 74 mm (8.7 x 10.7 x 2.9 inches) with the battery attached. The total weight of the system is 775 g (1.7 lbs) with no modifications and with the battery attached, as it was utilized for these experiments. This system is capable of a maximum flight time of 23 minutes with its single 3slp 4280 mAh LiPo-battery. This flight time provided ample in-air time for all experimental flights performed.

The key features used in this research were its ability to shoot 4k video at 60 Frames Per Second (FPS) as well as its native collision avoidance techniques. The high resolution and high frame rate provides a large amount of data to be post-processed for flaw detection which was done with the vehicle's primary camera, a Sony IMX577. The primary camera was used during all flights in a 90°gimballed down view to record the surface of the subject aircraft. The Skydio 2 employs mature collision avoidance algorithms utilizing 6 navigation cameras that allowed for a large degree of safety during flights around the scaled-aircraft and the full-sized aircraft owned by the museum.

The only modification to the Skydio 2 after purchase was the addition of 8 Infrared (IR) reflectors. The Vicon chamber utilizes IR light reflected off of these markers to detect the location of each reflector. Utilizing the Vicon Tracker software, an object was created for each test article that is built from the associated IR reflectors. The IR reflectors were placed in an asymmetric pattern on the Skydio 2 to ensure the Vicon system was able to determine the orientation of the vehicle as it rotated during flight. These reflectors are then tracked by the Vicon chamber cameras to determine the object's location relative to the origin, defined as the center, of the chamber.

The Skydio 2 uses a NVIDIA Tegra X2 System On Chip (SOC) for autopilot control as well as a NVIDIA Pascal Graphics Processing Unit (GPU) and NVIDIA Denver 2 Central Processing Unit (CPU) for video data processing. Communication with the vehicle was done using a Skydio 2 controller that communicates over WiFi in the following ranges: 2.4-2.483, 5.18-5.24, 5.725-5.85 GHz. This communication method allows for flight within 2 miles of the pilot but was only flown within a few hundred meters of the pilot during the trials.

3.1.2 Skydio 2 Systems Architecture

A systems architecture was created for the Skydio 2 to thoroughly document the relationship between the UAV components. Cameo Systems Modeler was used to create the systems architecture for this vehicle. The Physical Decomposition is shown in Figure 3.

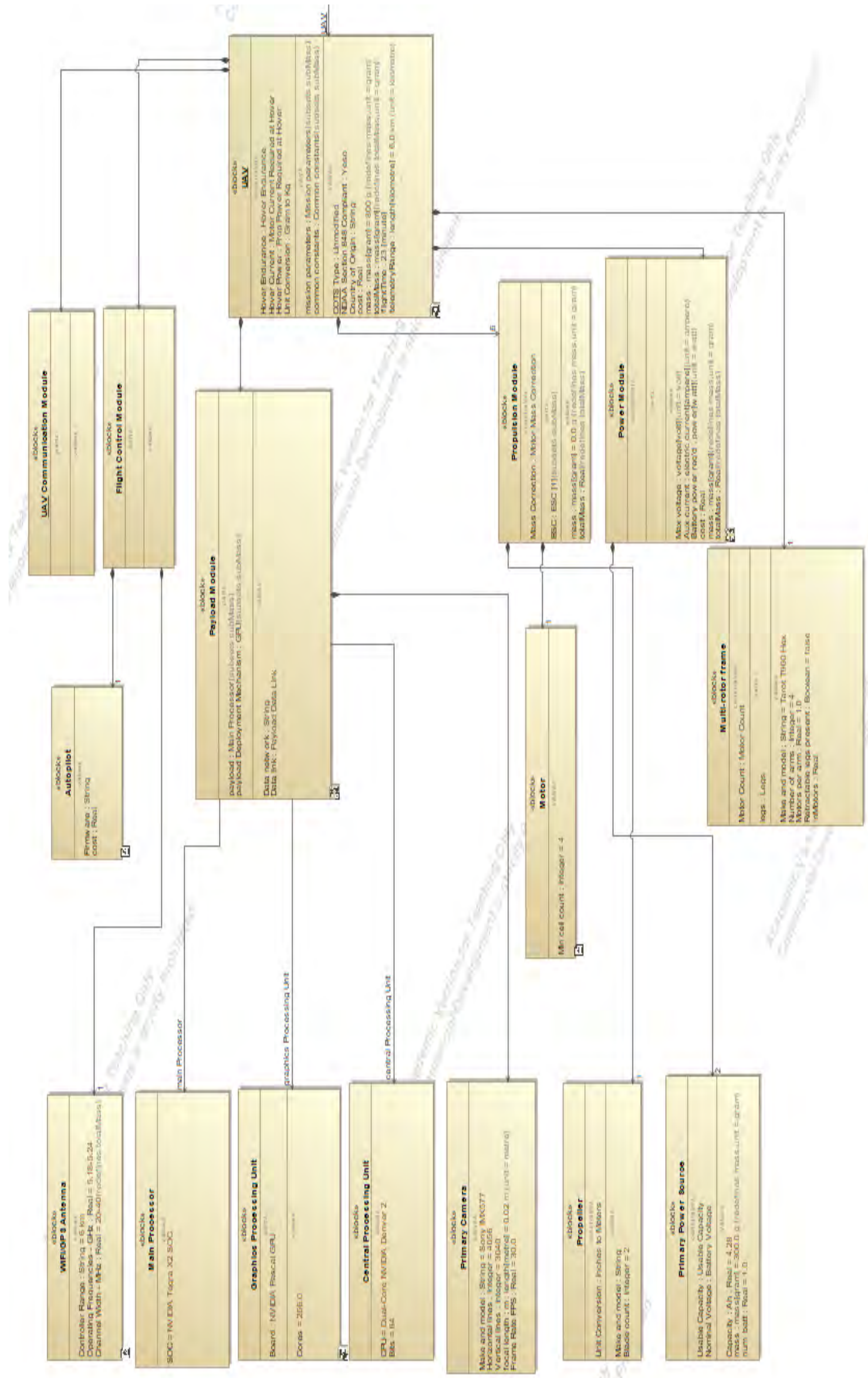

Figure 3:Skydio 2 Physical Decomposition

Due to the Skydio 2 being a COTS system this systems architecture was created with the publicly available list of hardware. Skydio does not include a full list of all components used in this system and without a complete teardown of the vehicle a fully accurate architecture could not be created. Therefore, these diagrams solely include the layout and the relationships of the major components.

3.1.3 Hexrotor

The Hexrotor utilized for this research was originally built by students within the AFIT Systems Engineering program. Figure 4 shows the modified Hexrotor as it was flown for these experimental flights.

Figure 4: Hexrotor with camera mount in Hangar

The Hexrotor was modified to include a 4k camera that records footage at 24 FPS. The bottom of the image frame in Figure 4 shows the camera mount as it was used during test. The camera payload was a Mako G-507B with a Sony IMX264 sensor.

The FOV of this sensor with a 3.5mm lens was much larger than the Skydio 2 FOV, and as such was mounted low under the vehicle to avoid capturing the vehicle's legs in the footage. The Pixhawk2 telemetry was used in flight to determine altitude of the vehicle via its barometer to enable more accurate tests.

The vehicle is capable of 22.2 minute flights with 2 10,000 mAh LiPo batteries. With the camera payload the total system weight is 7.91 kg (17.4 lbs).

3.1.4 Hexrotor Systems Architecture

A systems architecture was created for the original build of the Hexrotor used in this research by the AFIT sUAS (small Unmanned Aerial System) Track Class in 2021. This architecture was modified for this work to represent the changes to this vehicle. The changes include the removal of the stereoscopic camera and payload delivery system and the addition of the Mako G-507B.

Cameo Systems Modeler (CATIA) was used to create the systems architecture for this vehicle. The physical decomposition of the system is included to show the components required for the Hexrotor UAV as well as the GCS that was used to communicate with the vehicle. Portions of the Internal Block Diagram (IBD) are included for further documentation of the systems architecture.

3.1.4.1 Hexrotor Physical Decomposition

The physical decomposition for the Hexrotor displays the physical connections between each component of the UAV. This diagram provides a thorough overview of each component's key characteristics and specifications while indicating how each component interacts within the overall system.

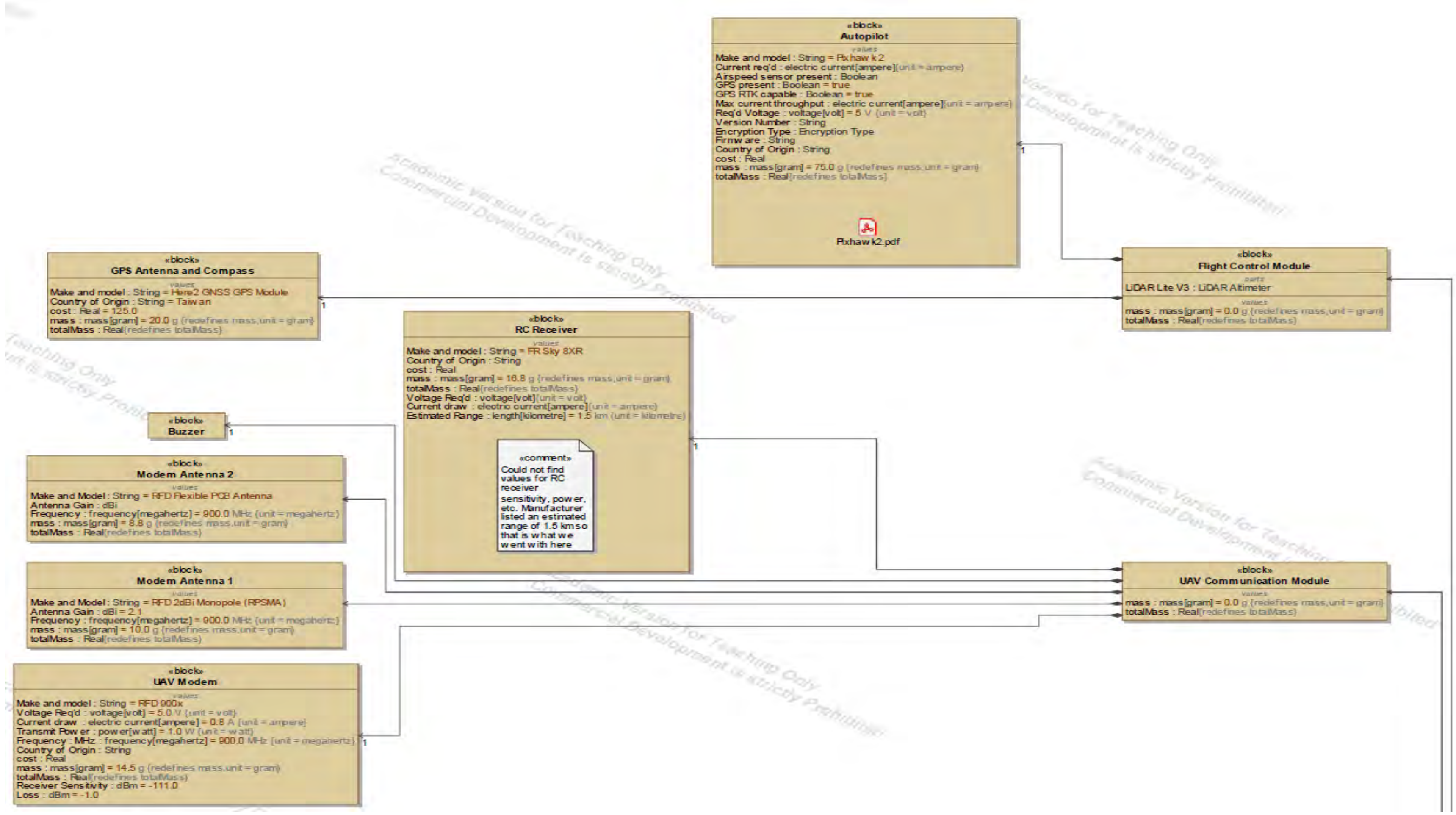

Figure 5:Hexrotor Physical Decomposition: Communication

The communication and flight control components of the Hexrotor Physical Decomposition is composed of the hardware enabling the UAV to transmit and receive commands while operational. This includes multiple antennae, a modem, a receiver, and the autopilot. These components combine into two different modules: Flight Control and UAV Communication. These modules then feed into the full system UAV block.

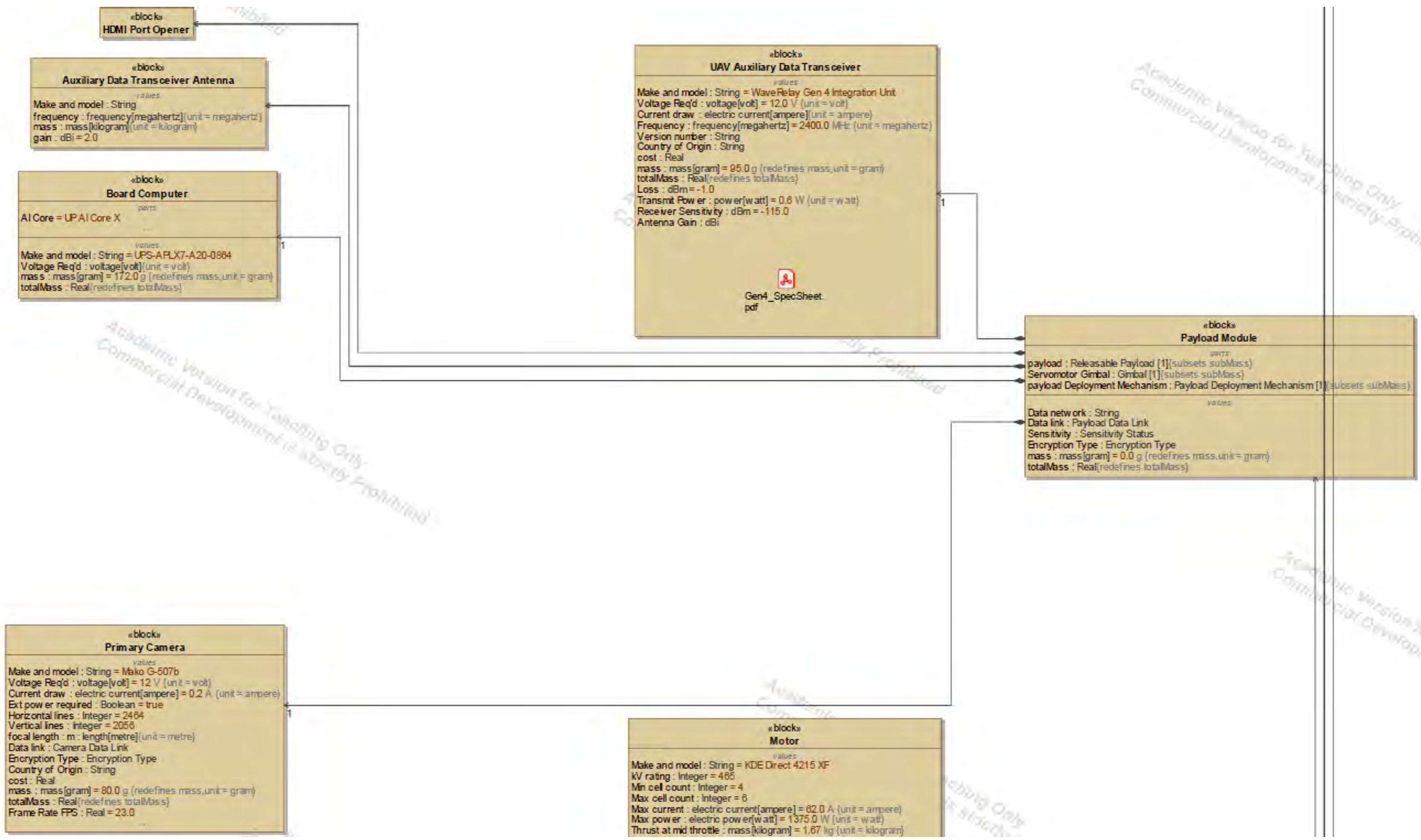

Figure 6:Hexrotor Physical Decomposition: Payload

The payload module decomposition is seen in Figure **??**. This payload includes the primary sensor, the Mako G-507B, as well as antennae to telemeter payload performance and communication data via the Auxiliary Data Transceiver. This information is not used for flight control of the UAV but enables the Ground Control Station (GCS) to issue commands to the recording aspect of the inspection.

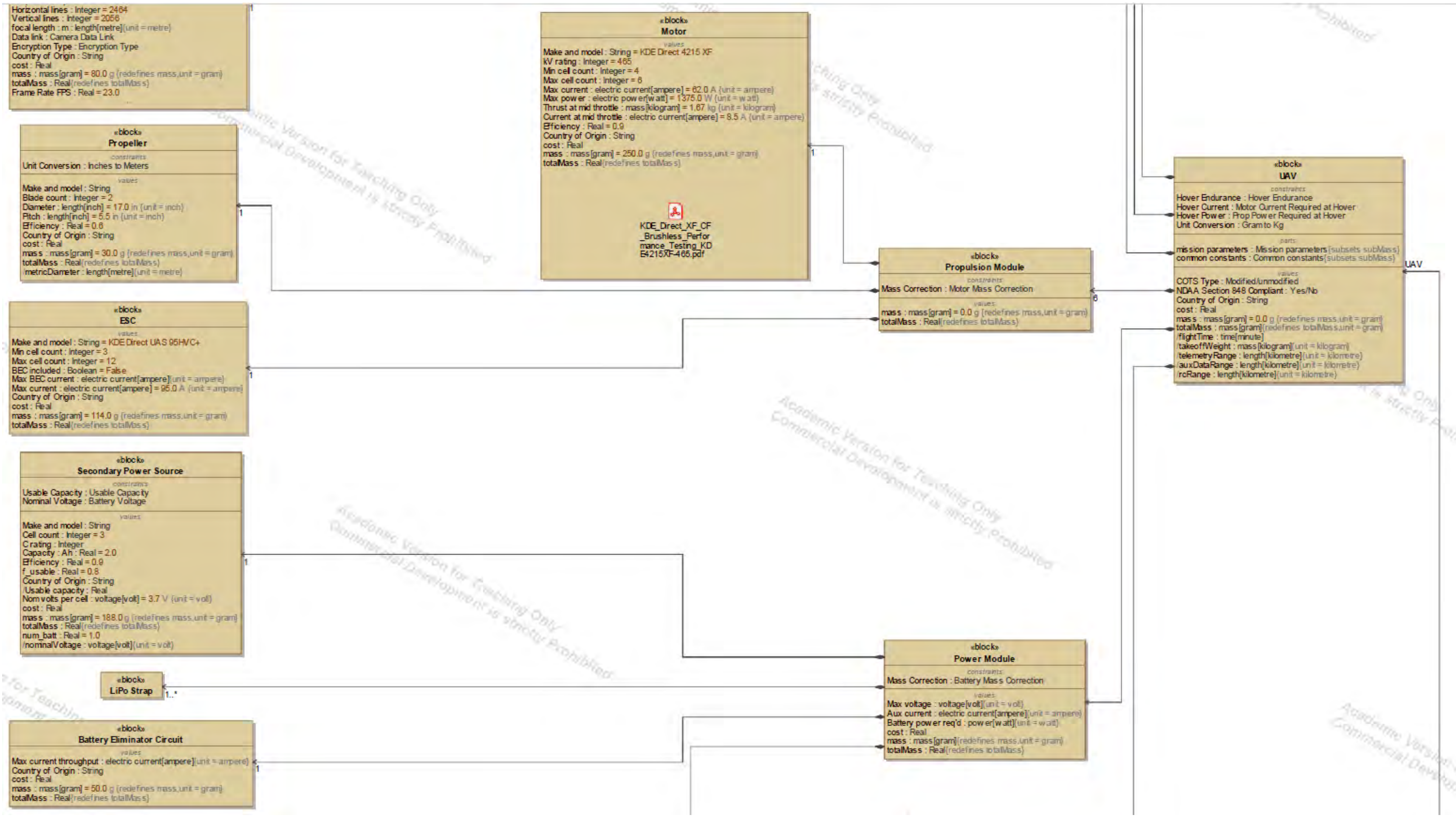

Figure 7:Hexrotor Physical Decomposition: Propulsion and Power

Figure 8:Hexrotor Physical Decomposition: Power and Frame

The propulsion and power modules depict the two modules that enable the Hexrotor to fly. These modules include the two 10mAh batteries providing power to the motors and all components on the aircraft. This section of the decomposition also displays the information about the system's motors and propellers.

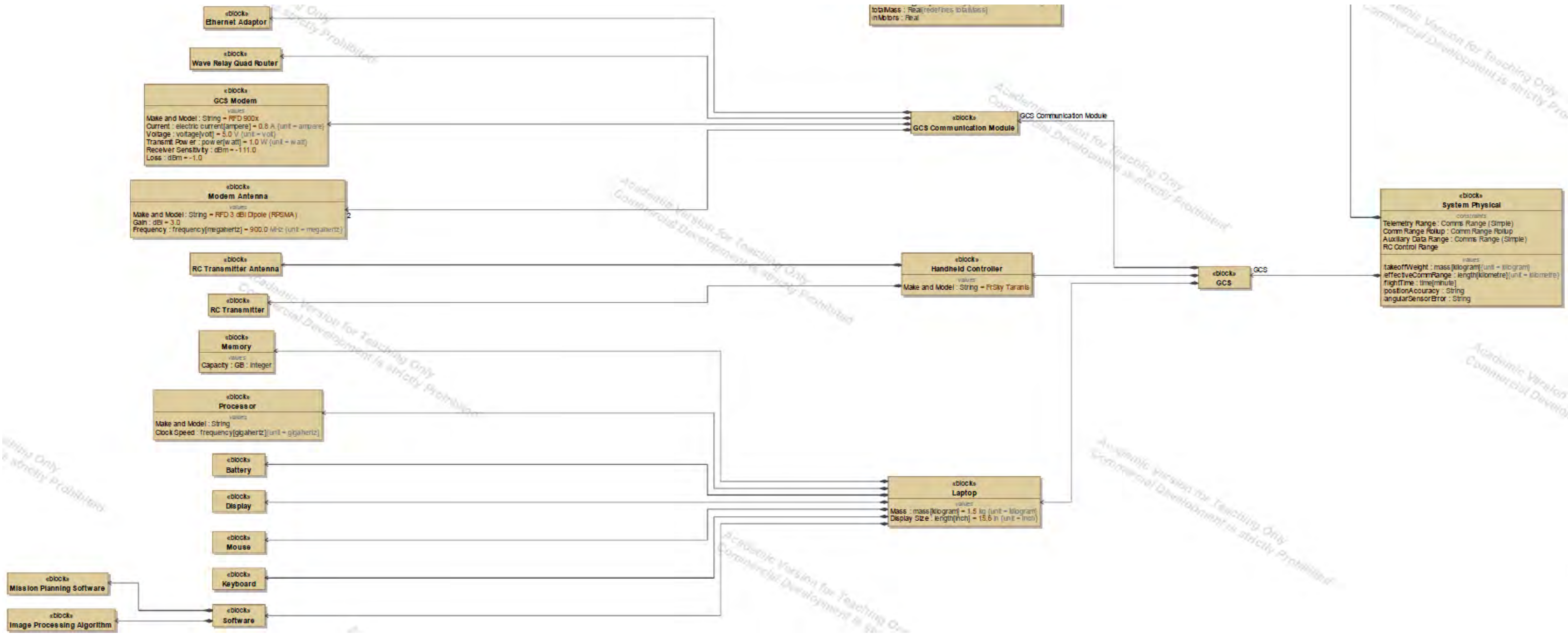

Figure 9:Hexrotor Physical Decomposition: GCS

The GCS module displays the interconnections between the laptop used for ground control as well as the handheld controller used to give flight control commands to the Hexrotor. A USB RC antennae module was connected to the GCS laptop to send and receive data from the autopilot component of the system which could be altered with the autopilot software.

3.1.4.2 Hexrotor Internal Block Diagram

The IBD for the Hexrotor was developed to display the connects between the components of the UAV. These connections show how key components such as the Pixhawk2, Mako camera, and communication system are interconnected to make an operable vehicle.

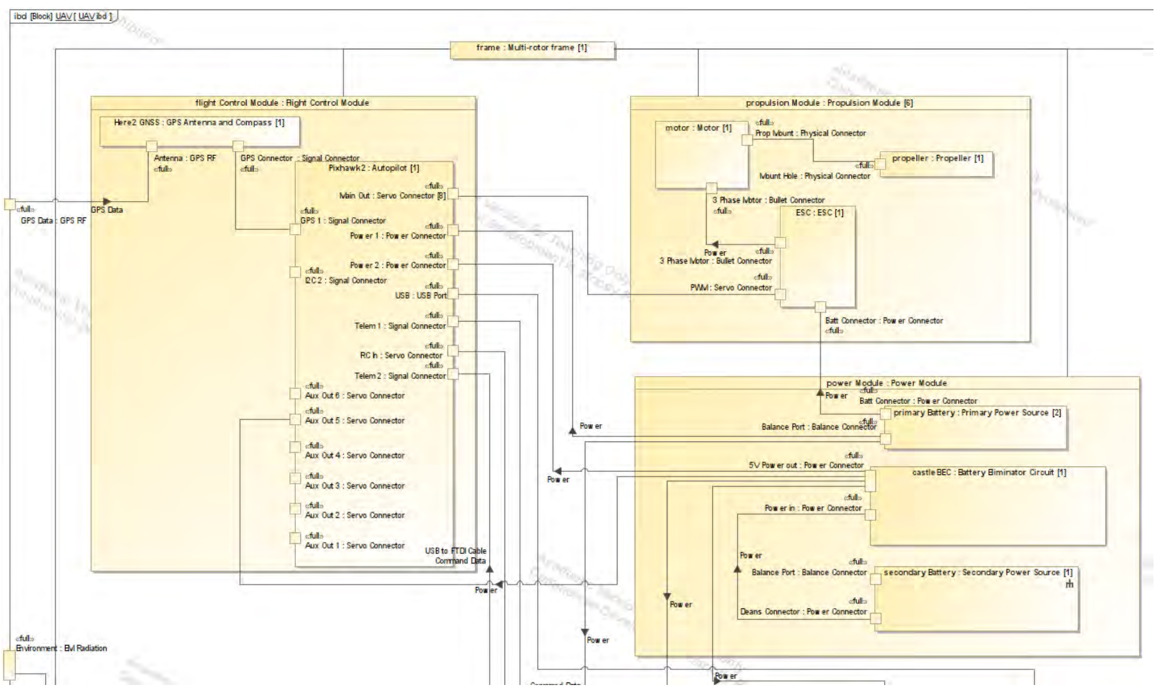

Figure 10: Hexrotor Internal Block Diagram: Flight Control, Propulsion, Power Modules

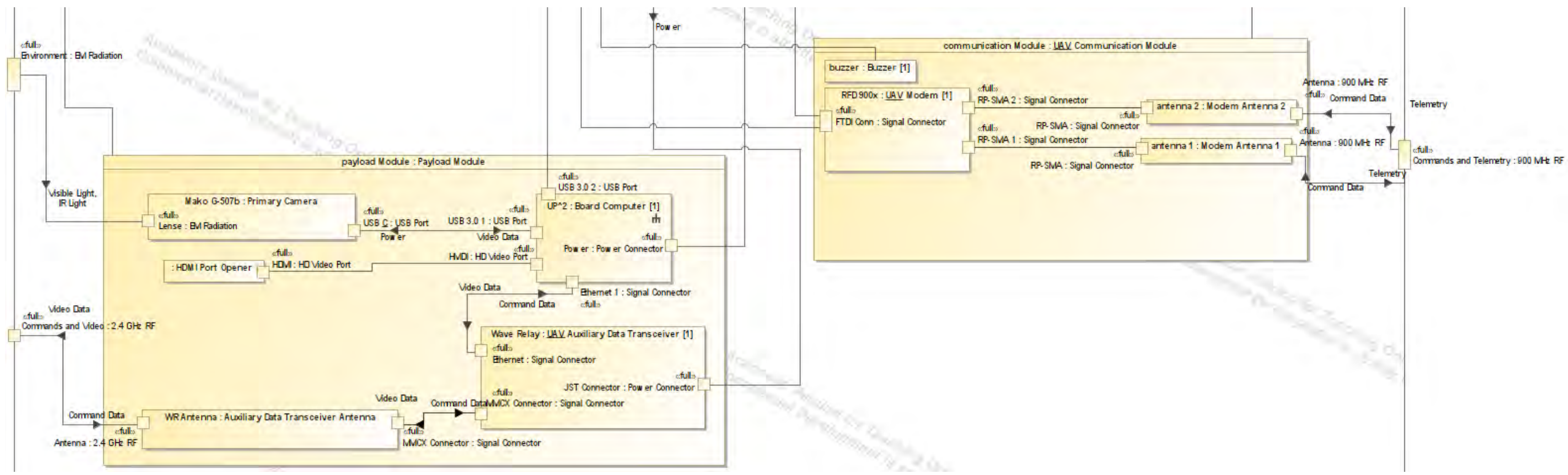

Figure 11:Hexrotor Internal Block Diagram: Payload and Communication Modules

3.1.5 FPS Considerations and Selection

When selecting the test articles with high resolution cameras it was the goal of this work to have systems equipped with 4k, 30 FPS sensors. This resolution was selected to provide the most pixels per flaw on the surface of the aircraft. The FPS was selected to achieve the highest amount of overlap possible in the recorded footage of the surface of the aircraft inspected. This frame rate allows for as close to full capture coverage of the inspected surface area of the aircraft while maintaining an inspection time of under 3 minutes for a full-sized vehicle.

3.2 Surface Flaw Pixel Analysis

An aircraft panel was received from the sponsor of this research to display examples of the flaws on the surface of an aircraft. This panel included examples of chips, scratches, and dents of varying sizes and shapes.

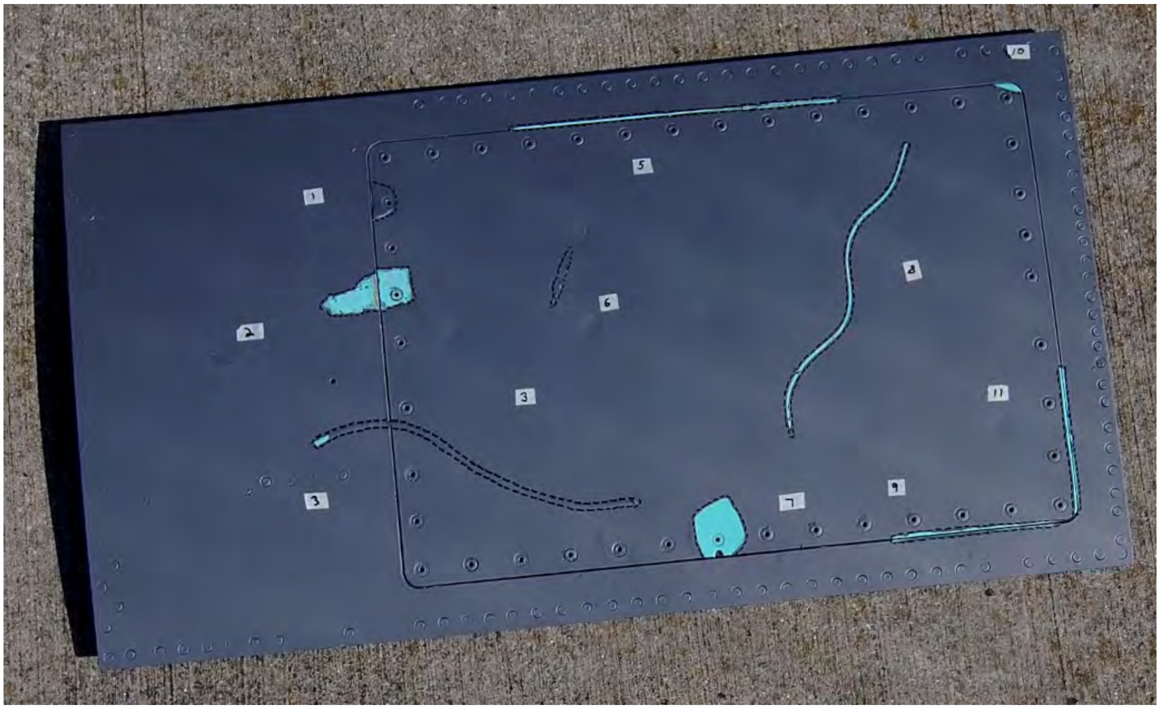

Figure 12: Aircraft Panel Zoomed in for Detailed Flaw View

The flaw examples were recorded with 4k images from 1m, 2m, and 3m heights to calculated the total pixels that each flaw took up within the image. The images these pixel dimensions for each flaw were calculated off of the following figures:

Figure 13: Aircraft Panel with Surface Flaws for Pixel Size Determination, 1m

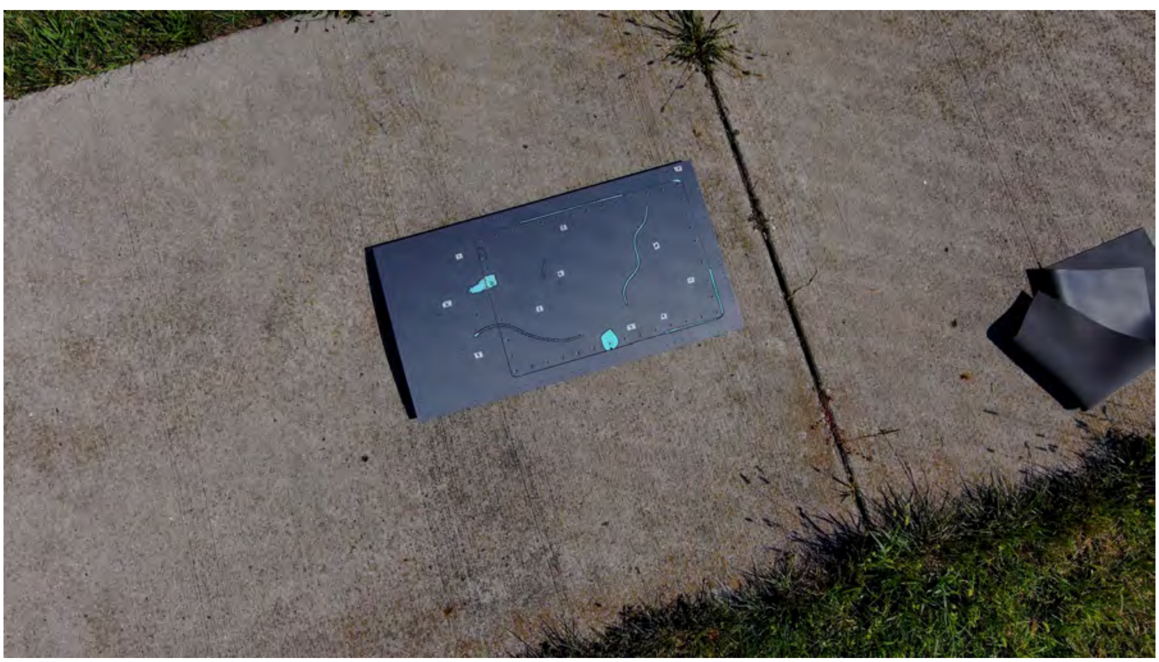

Figure 14: Aircraft Panel with Surface Flaws for Pixel Size Determination, 2m

Figure 15: Aircraft Panel with Surface Flaws for Pixel Size Determination, 3m

3.2.1 Panel Flaw Dimensions

The following table shows the total pixel area count for each scratch that are marked on the panel at each height:

Table 1: Flaw Pixel Area (Pixels Squared)(Marker 4 Unlabeled)

Trial Number	1	2	3	5	6	7	8	9	10	11
Scratch Size (mm)	2x3.5	8x3.5	27x1	28x1	1x6	4.5x5	27.5x1	14x1	2.5x1.5	13x1
1m	1254	5442	315	2200	778	4832	2886	2270	262	1990
2m	889	3493	132	1720	576	2650	1794	1183	183	956
3m	259	987	32	570	170	764	486	270	75	249

3.3 Flight Software

Previous research performed by the ANT center used Mako camera sensors similar to the one utilized on the Hexrotor in this research. Due to this dockerized software was used that the ANT Center was familiar with to run the camera while storing data in flight. The data was stored as Lightweight Communications and Marshalling (LCM) messages to log files. These messages log all ArduPilot telemetry data, Precise Time Protocol (PTP) timing data, and video footage. All data is then stored on a Solid State Drive (SSD) mounted on the frame of the vehicle. Having the PTP service running ensured that all data was stored with the same timestamp which enabled more accurate analysis.

3.4 Test Locations

Three different locations were used during the testing and experimentation of this research. All three locations are at Wright-Patterson Air Force Base (WPAFB)

3.4.1 AFIT Cage

The AFIT Cage is a located in the rear of the campus behind the ANT Center Lab. This is a large, outdoor space that is enclosed by netting. The Skydio 2 was flown within the cage to enable the safety pilot to become more familiar with the system. The Hexrotor was flown to verify that the addition of the Mako camera didn't cause the system to lose communication with the pilot, and the vehicle would fly as expected with the camera mount installed on the vehicle.

3.4.2 AFRL Indoor Flight Lab - Vicon Chamber

The majority of the data acquisition for this work was performed at the AFRL Indoor Flight Lab - Vicon Chamber. This space was used due to the IR camera motion capture capability in the chamber. This lab is a 25 x 20 x 10 m room. During all experimental flights the Vicon Tracker software saved the position and rotation information of the UAVs in flight. These UAVs were flown over a 1/7 scale F-15 model aircraft with an array of ArUco tags surrounding the model pictured in Figure 16.

Figure 16: 1/7 Scale F-15 Model with ArUco tags in Vicon Chamber

3.4.3 USAFM Restoration Storage Hangar

The final tests during this research occured at the United States Air Force Museum (USAFM) Restoration Storage Hangar. These experiments replicated the tests performed within the Vicon Chamber with a full-sized F-16 aircraft seen in Figure

17.

Figure 17: F-16 in the USAFM Restoration Storage Hangar

This location was used to perform the experiments in a real-world setting. The sponsor of this research hopes to be able to fly a UAV within a hangar over operational USAF aircraft. The F-16 that was recorded during these surface inspection trials was chosen due to its size being relatively similar to the fighter aircraft that the sponsor wishes to inspect.

3.5 Experiment Procedures

3.5.1 Flight Patterns

The intent of all experimental trials was to record the entire surface of each subject at varying heights with each flight while capturing ArUco tags within as many recorded frames as possible. The trials with both UAVs were performed at 1 m, 2 m, and 3 m from the surface of the aircraft. Maintainers are required to perform their visual inspections from at maximum a distance of 3 feet. To replicate this requirement, the closest flights were performed at 1 m (3.28 ft) from the surface. The flights performed over the 1/7 scale F-15 in the Vicon Chamber were done in 3 passes: one pass over the left wing, one down the center-line of the aircraft, and one final pass over the right wing. This flight pattern can be seen in Figure 18.

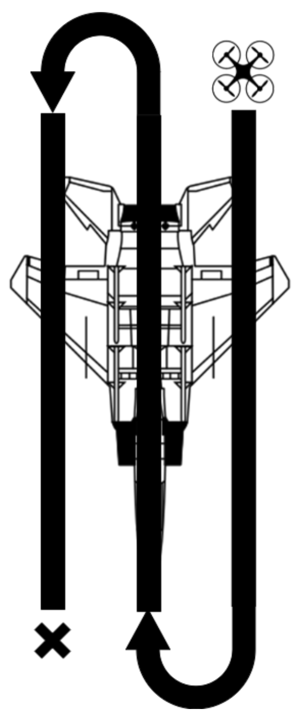

Figure 18: UAV Flight Pattern over F-15 Model

The flights performed over the full-sized F-16 in the Restoration Hangar varied with distance from the aircraft surface. Due to the size of the aircraft, if only three passes were performed at the 1 m and 2 m distances the entirety of the aircraft surface would not be captured. Instead, the UAVs were flown over each wing in two passes with a fifth pass over the center-line. These varying flight patterns can be seen in Figure 19.

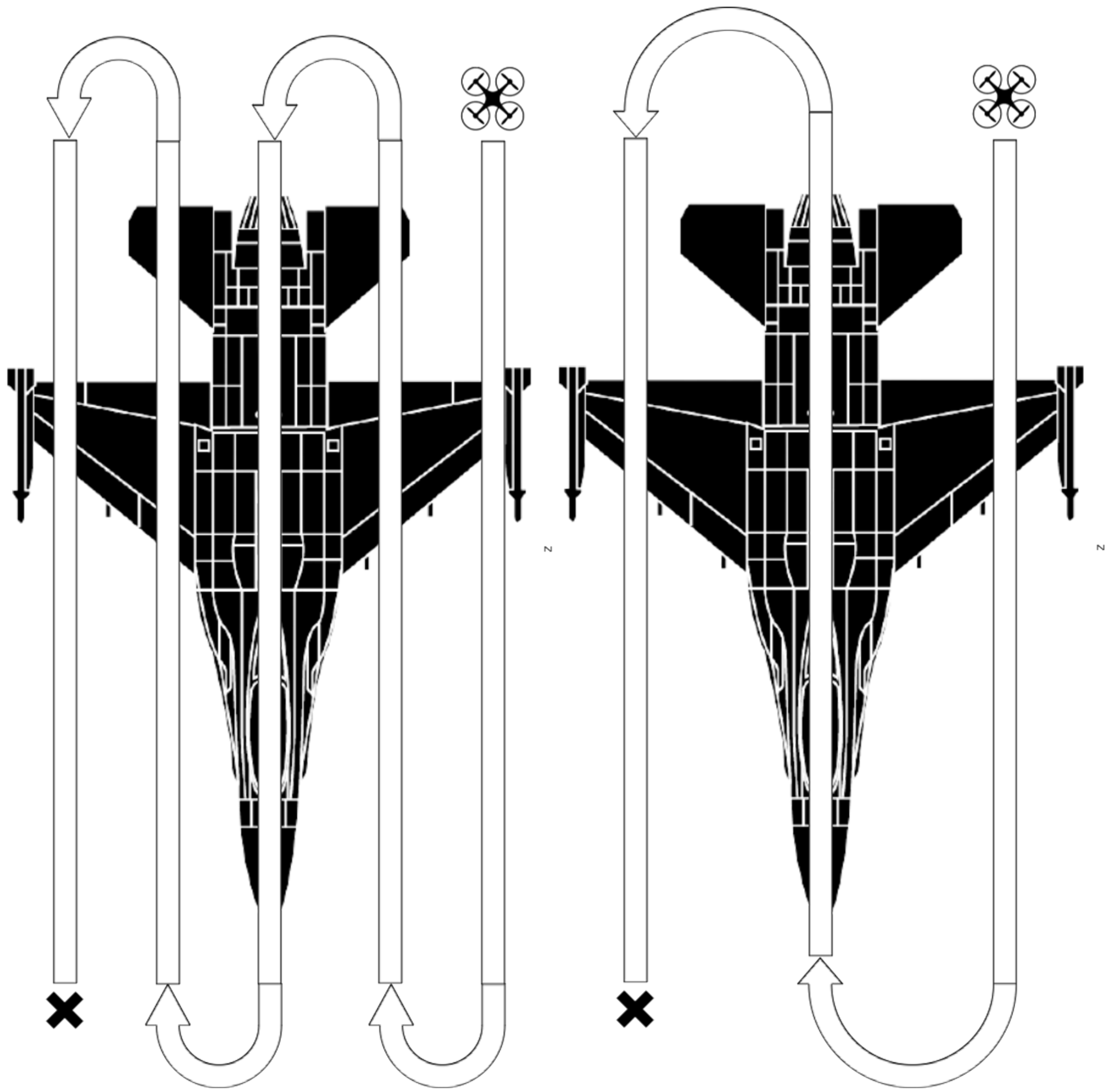

Figure 19: UAV Flight Patterns over F-16, Left: 1m and 2m Flights, Right: 3m Flights

Examples of the footage able to be recorded at the varying heights during the trials is displayed in Figure 20.

Figure 20: Left: Vicon Trials 1m, 2m, 3m Right: Hangar Trials 1m, 2m, 3m

This figure compiles a series of frames from trials performed with the Skydio 2 in both locations and at each trial height. The impact that the trial distance from the surface of the aircraft is easier to discern when observing the hangar trials. In a real-world setting it is clear that additional ArUco tags will be needed to capture more markers within each frame for pose accuracy and larger boards might be required as aircraft sizes vary.

3.5.2 Communication with UAVs

Figure 21: Hangar GCS, UAV communication Setup

The Skydio 2 communicated exclusively through its phone-based software connected to the Skydio Controller. The Hexrotor used a handheld controller that communicated directly to the vehicle. The vehicle and GCS communicated via Mission Planner to the Pixhawk2 used for navigation on the vehicle. The setup used for GCS can be seen in Figure 21. that communicated . In addition, a second computer communicated to the vehicle through the on-board wave relay to give the Intel Atom commands to start the PTP timing service, activate the camera, and to begin logging all data from the camera, PTP timing service, and Pixhawk2 data.

3.5.3 ArUco Tag Positioning

ArUco tags were used in this research to determine the location of the UAVs in flight. Recordings of the surface of the aircraft inspection captured flaws on the aircraft surface as well as ArUco tags positioned on the floor. The positions of the ArUco tags were chosen to increase the likelihood of multiple tags being capture within frames recorded during inspection flights while simultaneously capturing the surface of the aircraft. The layout of the tags varied with the scale of the subject aircraft. These layouts are seen in Figure 22 and Figure 23.

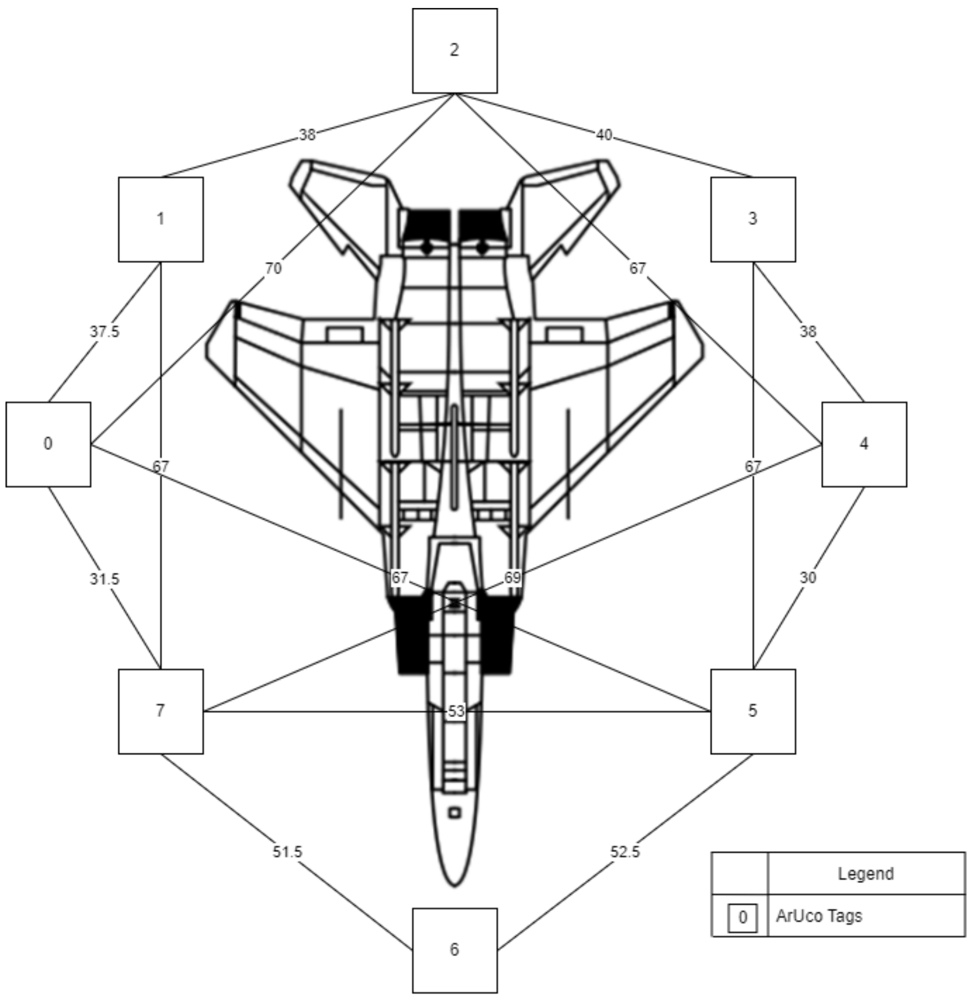

Figure 22: F-15 ArUco Tag Layout in Inches

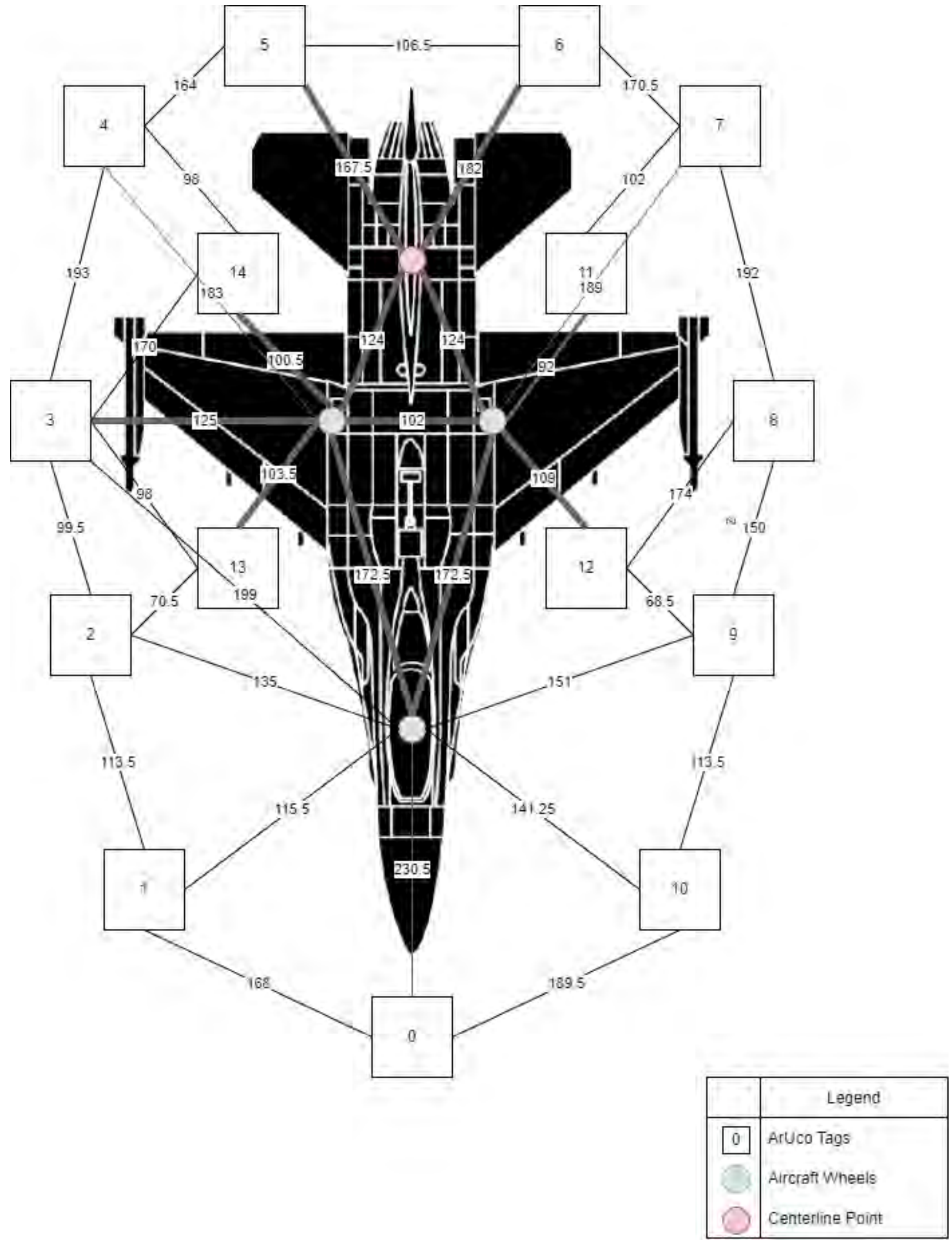

Figure 23: F-16 ArUco Tag Layout in Inches

The positions of the center of each ArUco tag surrounding the F-15 model were documented for a comparison between the position accuracy of ArUco positioning versus Vicon positioning. These Vicon positions are included in Table 2

Table 2: F-15 Model ArUco Tag Vicon Locations (mm)

ArUco Tag	X	Y
0	64.195	-922.161
1	-865.821	-731.449
2	-1425.525	48.802
3	-890.637	912.185
4	72.524	865.176
5	785.199	624.808
6	1946.106	-14.807
7	835.247	-721.189

3.5.4 Detection and Processing Scripts

The creation of scripts to analyse the inspection followed an iterative process. OpenCV and Python libraries are open source and were utilized to create the required scripts. Due to the open source nature of these libraries there are many examples to use as a starting point. [22][28] The python code created to perform the analysis of this work will be listed and publicly accessible in this research's GitHub space. [29]

One script was created to complete all data processing, pose estimation, and data comparison between the ArUco and Vicon systems. This script is entitled poseEstCamera.py. and was created to process the recordings of the aircraft surface inspections.

The poseEstCamera.py utilized code from Automatic Addison's [28] based on the OpenCV tutorials on camera calibration, usage of calibration files, and detection of ArUco tags. This script expands on these tutorials and the guide created by Addison by storing the pose estimations of the ArUco tags seen in each video frame, the estimated pose of the UAV's camera, and using this information to calculate extrinsic values from the pose estimation with this data. During this process the pose of the ArUco tags and the camera are stored as vectors. The calculated pose of the camera is then shifted to be at the center of mass of the UAV, rotated into the Vicon chamber frame, and the extrinsics error calculated and documented within graphs. An example of the ArUco pose estimation is seen in Figure 24. In addition to the ArUco related functions outlined in this chapter, the OpenCV Rodrigues function intakes a rotation matrix and outputs a set of rotation vectors and translation vectors built from the rotation matrix data which was used throughout both methods.

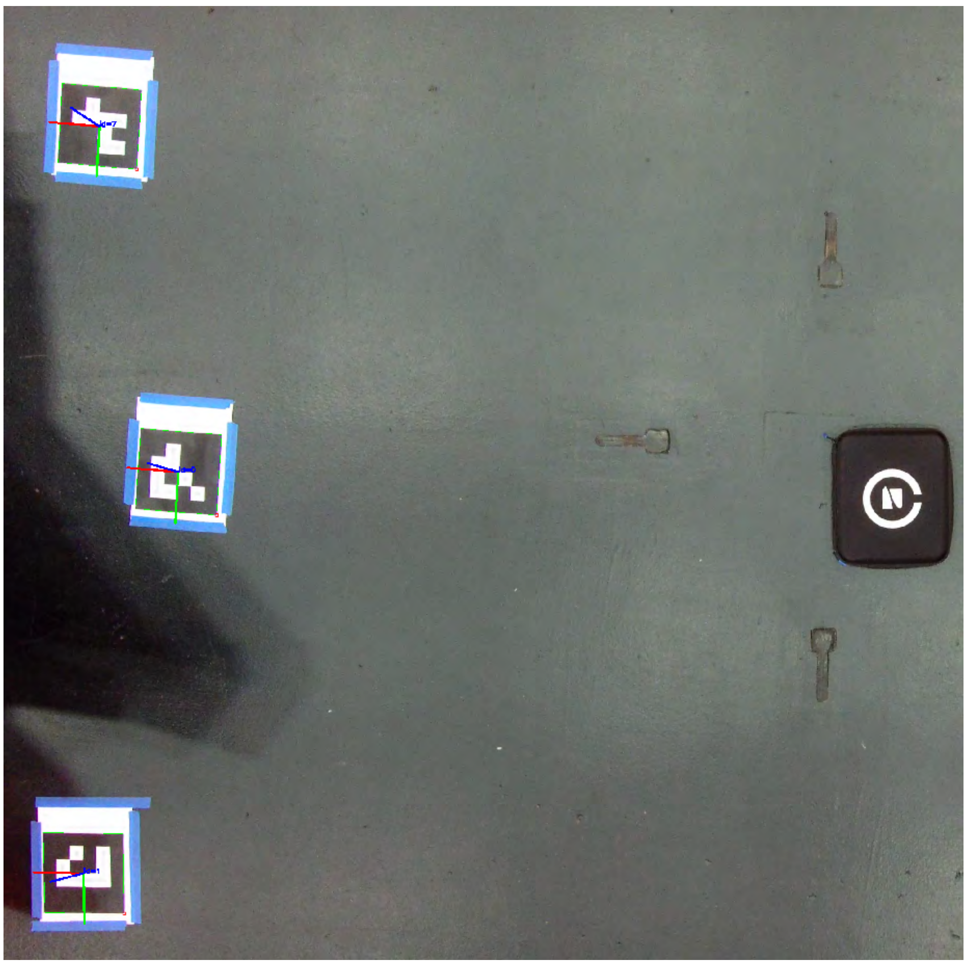

Figure 24: Skydio2 Trial Footage with ArUco Tag Pose Estimation

3.5.4.1 Individual Marker Detection

Figure 25 displays the rotations that were done by the code to calculate the extrinsics of the camera from the OpenCV marker frame to the Vicon frame to the Aircraft frame and finally to the camera frame. In the poseEstCamera.py file the rotations are performed in this order although the rotations and translations could be performed in the opposite direction, starting from the Aircraft frame and ending once again at the camera frame. Equation The following equation was used to determine the estimated translation between the camera and the UAV,

$$^{om}\hat{\mathbf{t}}_c = R_v^{om}(-\,^{v}t_m + \,^{v}t_q + R_{vq}^{v} \cdot \,^{vq}t_c) \tag{1}$$

This displays the math performed to determine the estimated translation from each individual OpenCV Marker position to the position of the camera. The equation is explained by the following while referencing the translations and rotations depicted in Figure 25. $^{om}\hat{\mathbf{t}}_c$ indicates the estimated translation from the camera to the OpenCV marker frame, black. R_v^{om} indicates the rotation from the OpenCV marker frame to the marker frame, yellow. $-\,^{v}t_m$ indicates the translation from marker to vicon frame, yellow. $^{v}t_q$ indicates the translation from the vicon to UAV frame. R_{vq}^{v} is the rotation from the vicon frame to the UAV frame. $^{vq}t_c$ is the translation from the UAV frame to the camera frame.

This information was used to calculate the A and B matrices required to find the extrinsic translation between the camera and the UAV frame. The A and B matrices are found in Equation (3):

$$^{om}t_c + R_v^{om}\,^{v}t_m - R_v^{om}\,^{v}t_q = R_v^{om} R_{vq}^{v}\,^{vq}t_c \tag{2}$$

The B matrix is made up of the left portion of Equation (3), $^{om}t_c + R_v^{om}\,^{v}t_m -$

$R_v^{om\,v}t_q$. The B matrix performs all the rotations and translations necessary to move from the Camera frame to the UAV frame as shown in Figure 25. The A matrix consists of the following portions of Equation (3): $R_v^{om} R_{vq}^v$

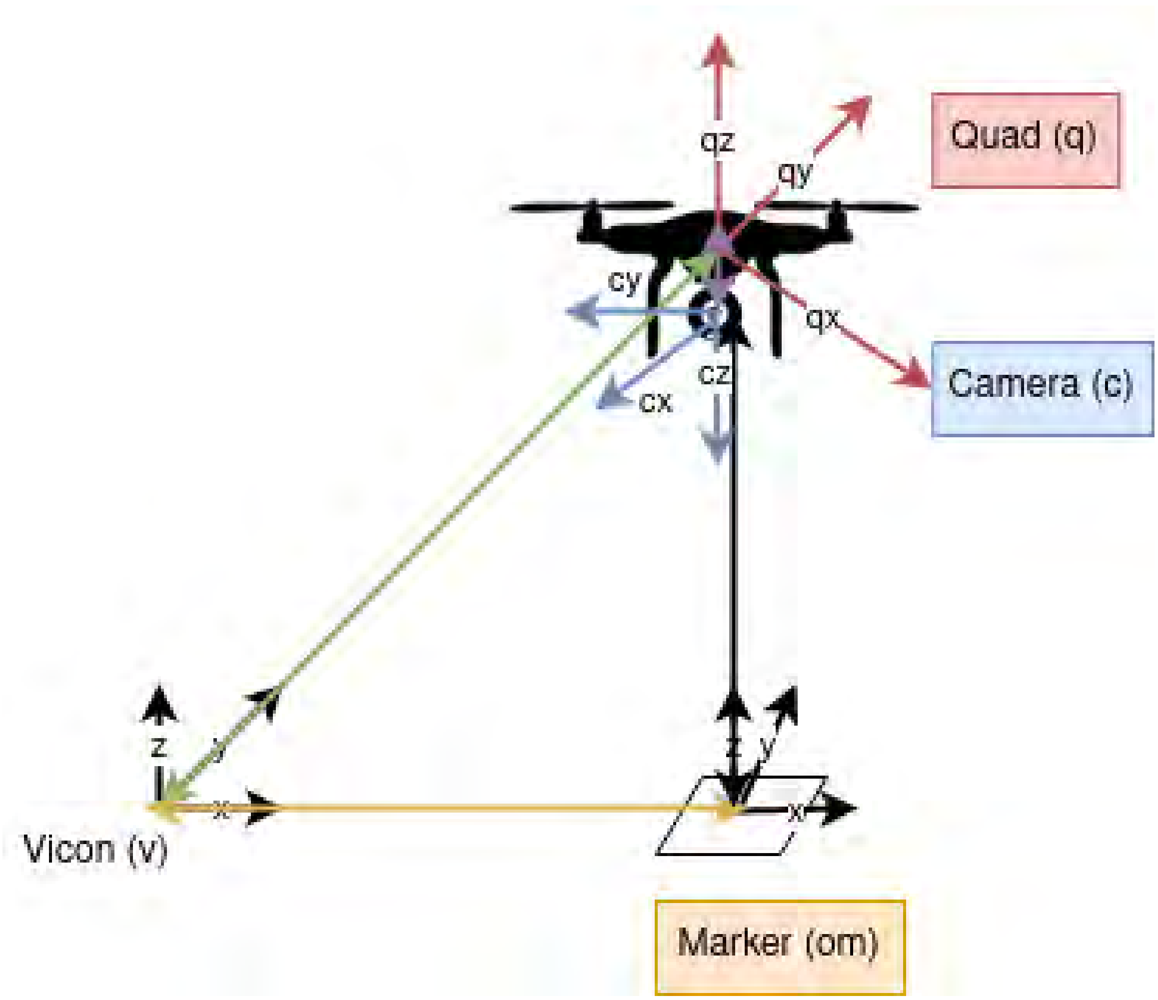

Figure 25: Rotations to Determine Camera Extrinsics

3.5.4.2 Perspective N- points Problem (PnP) Usage

During the process of calculating the B matrix error, see in Section 3.5.4.1, it was found that the position estimation based on the static Vicon-detected corners of the ArUco tags along with the aruco::estimatePoseSingleMarkers() function was not producing an accurate enough pose estimation. Due to the inaccuracy of the pose estimation, determining an accurate position of the UAV in-flight was very unstable. To combat this unreliability OpenCV PnP functions were used. The cv:solvePnPGeneric() function uses the world frame 3D points alongside the image frame 2D points to more accurately estimate the pose of the camera recording the imagery. The general use of solvePnPGeneric as performed by OpenCV is seen in Figure 26 which was also discussed in Chapter II.

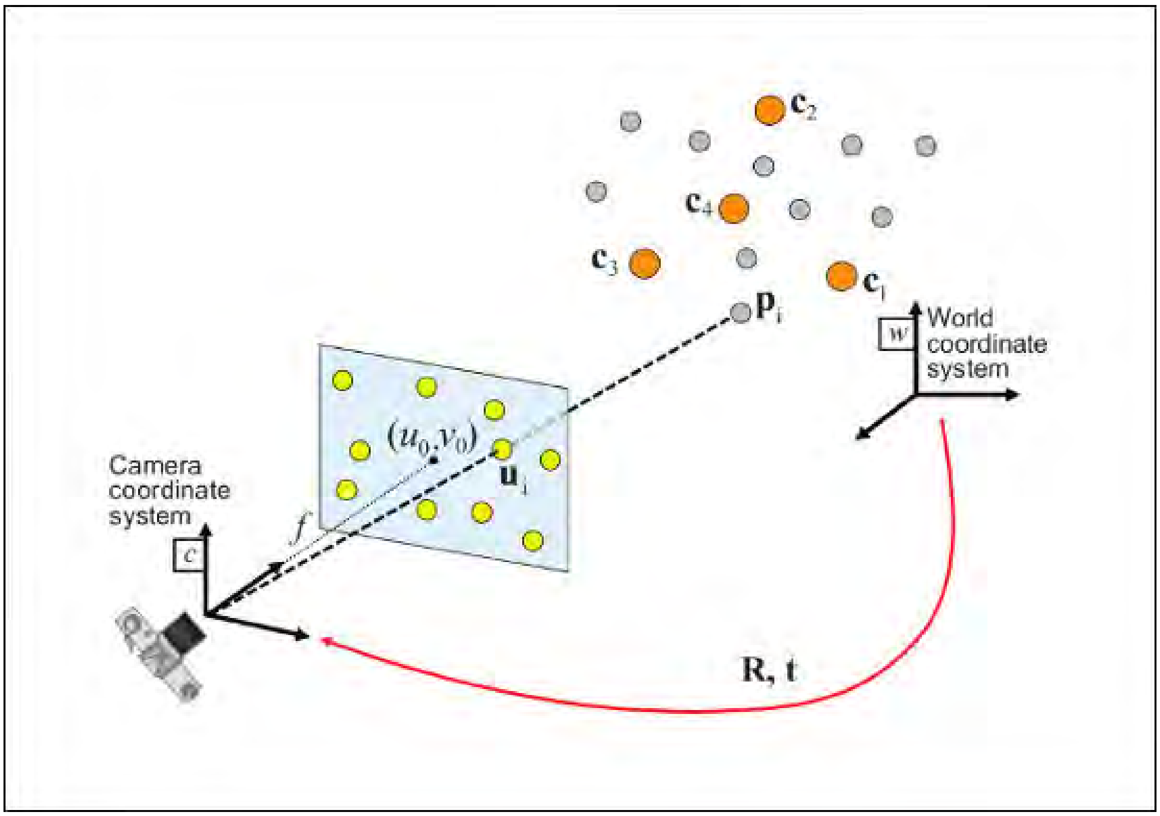

Figure 26: OpenCV: Perspective N-point Problem[1]

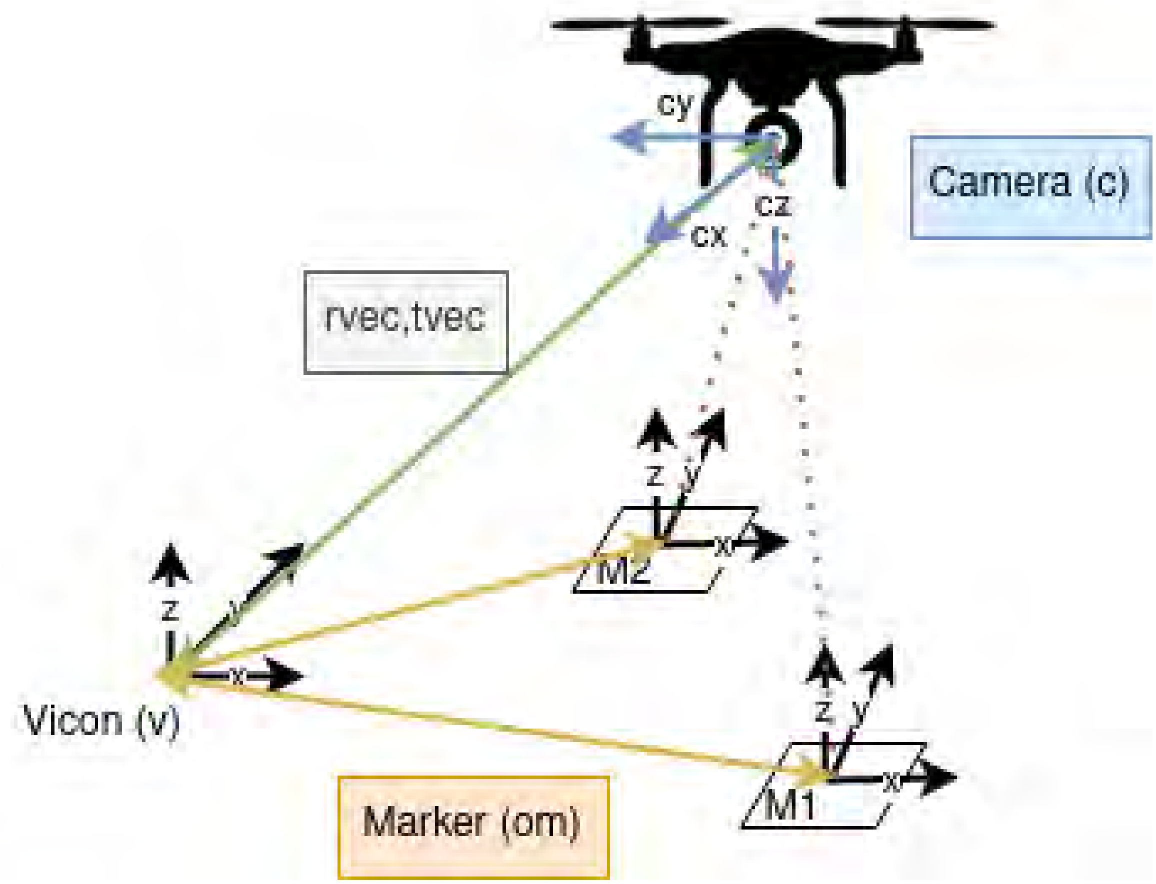

Figure 27: PnP Use Case Demonstration

Figure 27 visualizes the way that solvePnPGeneric was used in this work's research. The world frame 3D object points of the markers, as detected by the Vicon system, were utilized alongside the 2D image points, defined as corners by the ArUco tag detection functions. These 2D and 3D points were used to determine rotation (rvecs) and translation (tvecs) vectors in the Camera frame. This PNP method determined these rvecs and tvecs utilizing all detected markers within a single frame simultaneously for a pose estimation as opposed to the method displayed in Figure

25 which performs and documents pose estimation with individual markers.

3.5.4.3 Data Note:

As LCM was utilized to store the data from the Hexrotor flights all data was stored as binary files. Originally the data that was recorded from the Hexrotor video trials was going to be processed as well but due to shortcomings on time and position inaccuracies this analysis will be performed in future work.

IV. Results and Analysis

Preamble

This chapter is organized by first discussing the experimental results in section 4.1. This section deals with the graphs and accuracy findings of each pose estimation technique that was used during this research. Section 4.2 discusses the ability to detect flaws from varying heights as seen by the trials conducted during the experiment portion of this research. This discussion focuses on imagery of example flaws that were placed on aircraft and the viability of the sensors used in this work. The final section of this chapter analyzes the Unmanned Aerial Vehicle (UAV)s used to perform these experiments and the advantages and disadvantages of each vehicle and future desired traits for vehicles performing similar experimentation.

4.1 Experimental Results

Nine total trials at the heights of 1m, 2m, and 3m were performed by each aircraft within the Vicon Chamber and the United States Air Force Museum (USAFM). In the case of the Vicon chamber trials 8 total ArUco tags were utilized leading to a maximum of 8 B matrices, defined in Section 3.5.4.1, being created during each inspection frame throughout each trial. The total frames that were analyzed by trial and cases where some markers were never detected during the duration of a trial are annotated in Table 3. The trial results are discussed by method in the following subsections and include plotted data at high resolution that contain greater detail when zoomed in.

Table 3: Vicon Trials: Frames, Clip Lengths, Markers Not Detected

Height	1m			2m			3m		
Trial	T1	T2	T3	T1	T2	T3	T1	T2	T3
Length(s)	21.4	18.1	26.7	10.4	12.5	17.4	10	7.3	6.5
Frames Used	642	544	801	312	376	523	300	218	197
Not Detected	0,6						6	6	6

4.1.1 Individual Marker Pose Determination

Individual Marker Pose Determination was discussed in Section 3.5.4 and is visualized in Figure 25 within Chapter III. The analysis performed on this pose estimation technique was focused around the error calculated by the B matrix described here.

$$ {}^{om}t_c + R_v^{om}\,{}^{v}t_m - R_v^{om}\,{}^{v}t_q = R_v^{om} R_{vq}^{v}\,{}^{vq}t_c \tag{3} $$

This error matrix is defined as the left portion of (3), which was derived from (1) as a way to calculate the translation from the camera to the UAV body, ${}^{vq}t_c$. The B matrix that was created to determine this translation had greatly varying results and is shown in Figure 28 and Figure 29.

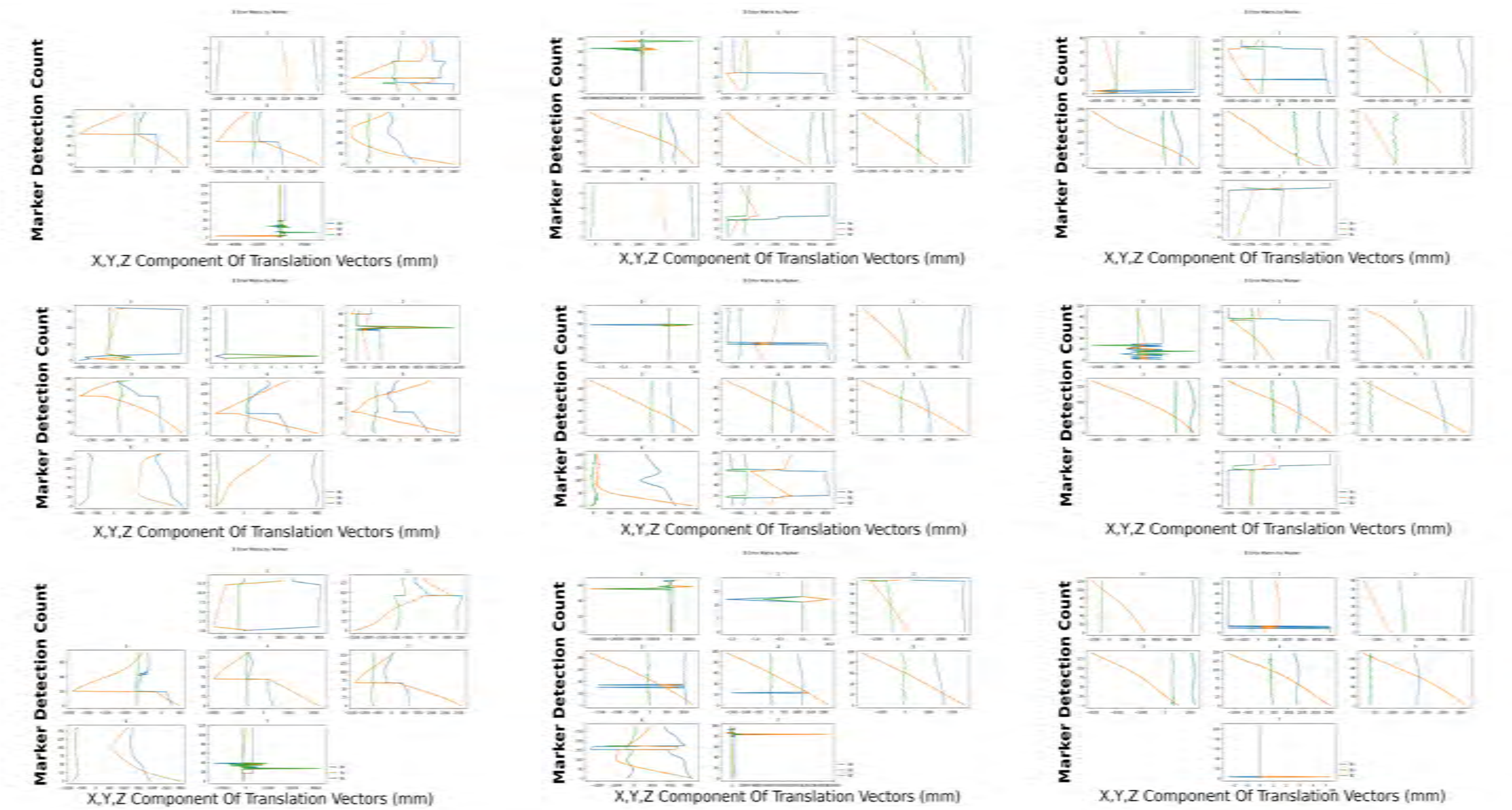

Figure 28:B Error Matrix Calculated for Each Trial and Marker

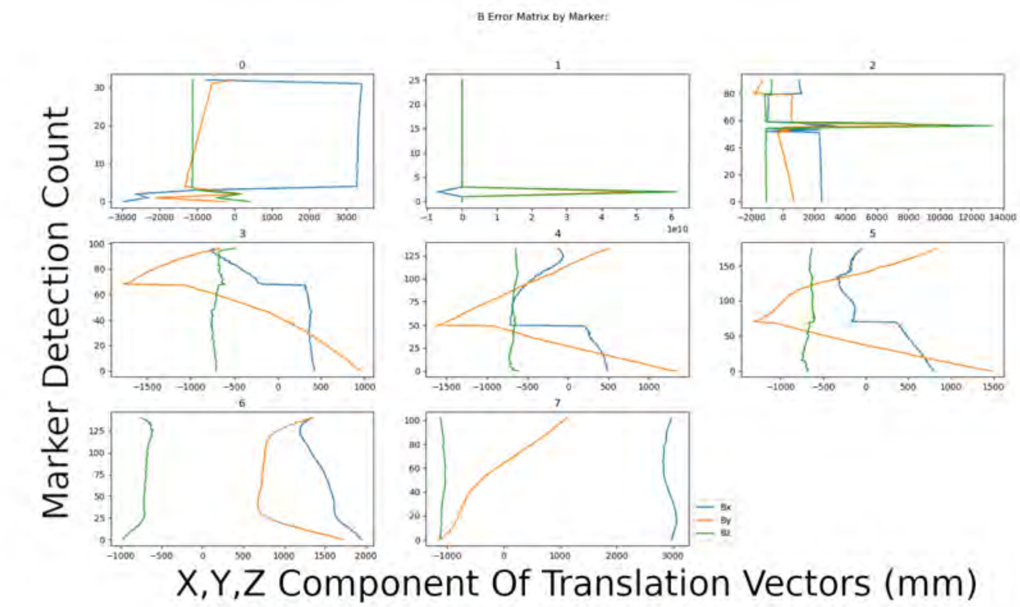

Figure 29: Example B Error Matrix Calculated for 1m Height Trial 2, middle of first row in Figure 28

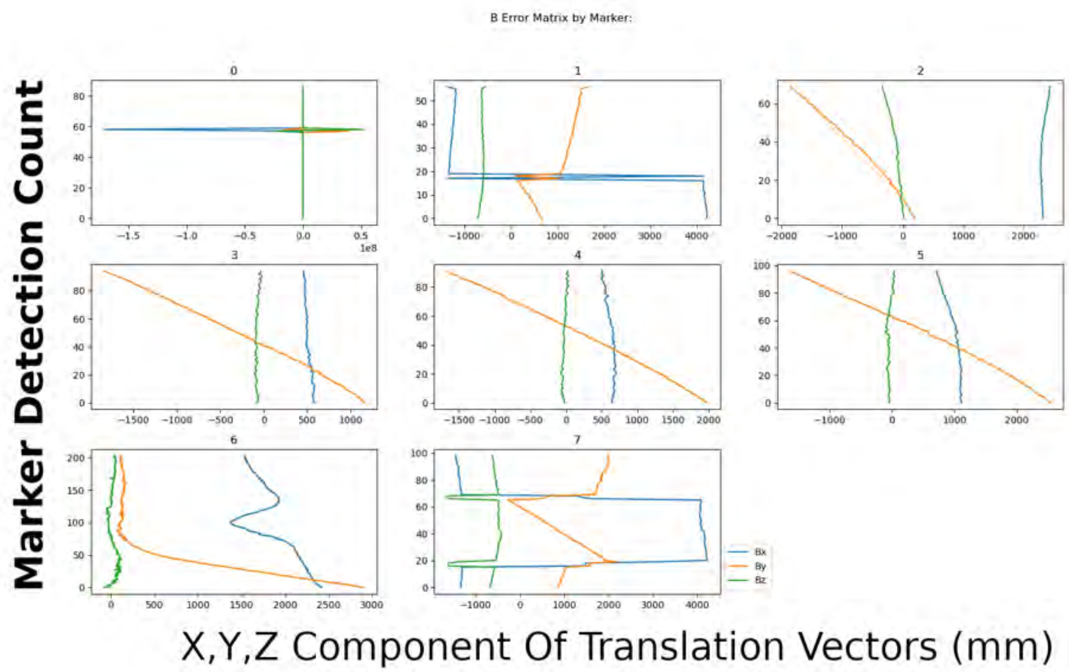

Figure 30: B Error Matrix Best Result During 2m Height Trial 2, middle of second row in Figure 28

Figure 30 performed the best for the individual pose estimation method. This trial was the second trial performed at the 2m height about the F-15 model and had the lowest change in the process of calculating the X, Y, and Z components relative to each marker.

The resulting B Matrices from this process vary greatly in their accuracy. All graphs display the total count that each marker was detected during the course of each trial performed and the X, Y, and Z component of the calculated translation. This translation indicates the estimated distance between the physical center of the UAV, not the center of gravity, and the camera attached to the vehicle as calculated off of each individually detected marker. As this position solution is only being calculated by an individual marker when detected in every frame the results varies greatly over the course of each trial. In addition to this, the estimates are displayed in millimeters and with errors in pose estimation are seen to determine the camera is multiple meters from the center of the UAV while in reality, the camera is a few centimeters from the center of mass of the vehicle.

This single marker pose estimation could be improved including tolerances for the distances that would be accepted as within a meaningful distance to the expected values the translation from camera to UAV body should be. However, due to the large jumps in distances detected from this method it is likely that a completely different method, such as PNP discussed in the next section, will always defeat a method that performs detection relying on individual markers.

4.1.2 Solve PNP Pose Determination

The second function used to perform pose estimation in this research was solvePnPGeneric(). This function is visualized in Chapter III in Figure 27. As described in the Methodology chapter, this function utilizes all markers detected in an individual frame and uses the corner of each marker to estimate the pose of the camera relative to these markers and the Vicon frame. This eliminates the majority of the rotations that were necessary to determine the camera pose with the individual marker pose estimation method. After determining the pose of the camera frame, the translation vector, ${}^{v}t_{c}$, and rotation matrix, R_c^v, to move from the camera frame to the Vicon frame was determined using Rodrigues(). This translation vector and rotation matrix were then used in combination with the results from the individual method, ${}^{v}t_{vq}$ and R_{vq}^{v} , to create a skew symmetric matrix. This is created by the translation vectors seen in Figure 31.

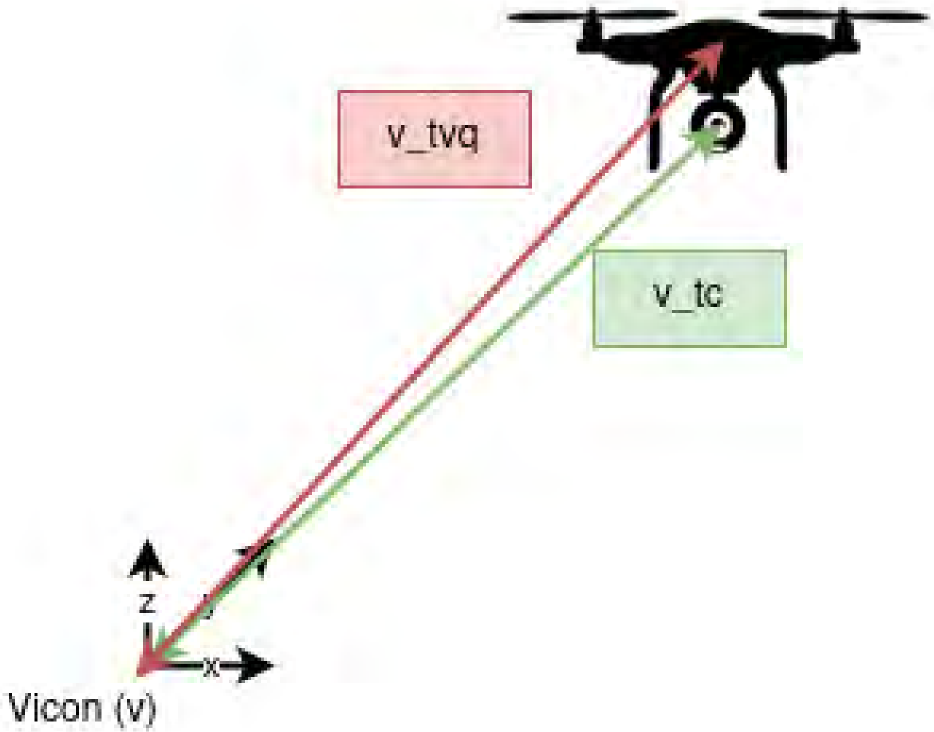

Figure 31: Visualization of the Skew Symmetric Matrix

The resultant skew symmetric matrices were calculated by 4 to determine the difference in error between the two translations from the Vicon frame to the quad and camera frames

$$SkewSymmetricError = I - {R_{PNP}}_v^{vq^T} R_v^{vq} \tag{4}$$

.

The plotted results of the skew symmetric matrices created over the course of each trial within this experiment are included below and display the error for the X, Y, and Z rotational components calculated over the course of each trial:

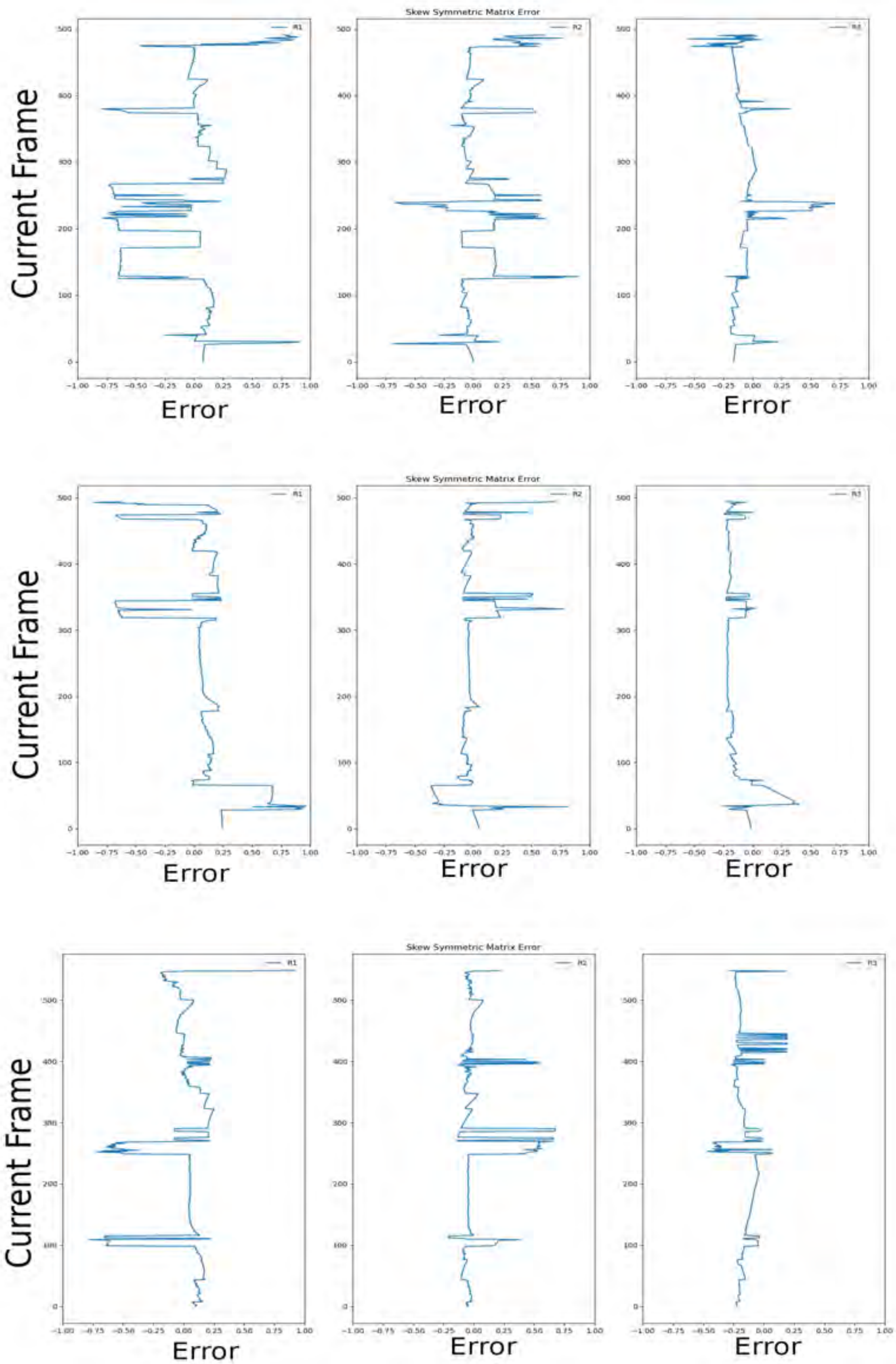

Figure 32: Skew Matrix Results for 1m Skydio Trials

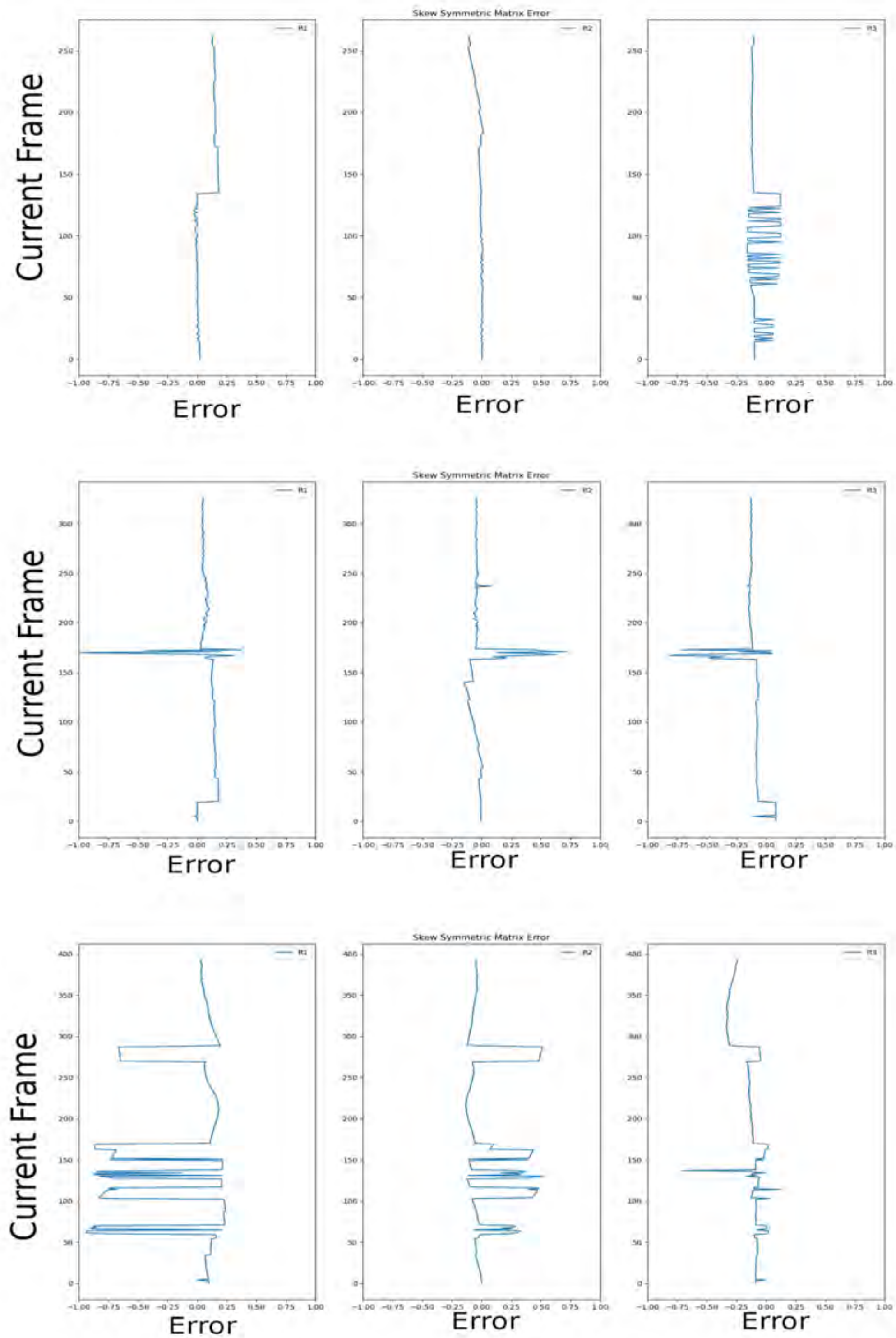

Figure 33: Skew Matrix Results for 2m Skydio Trials

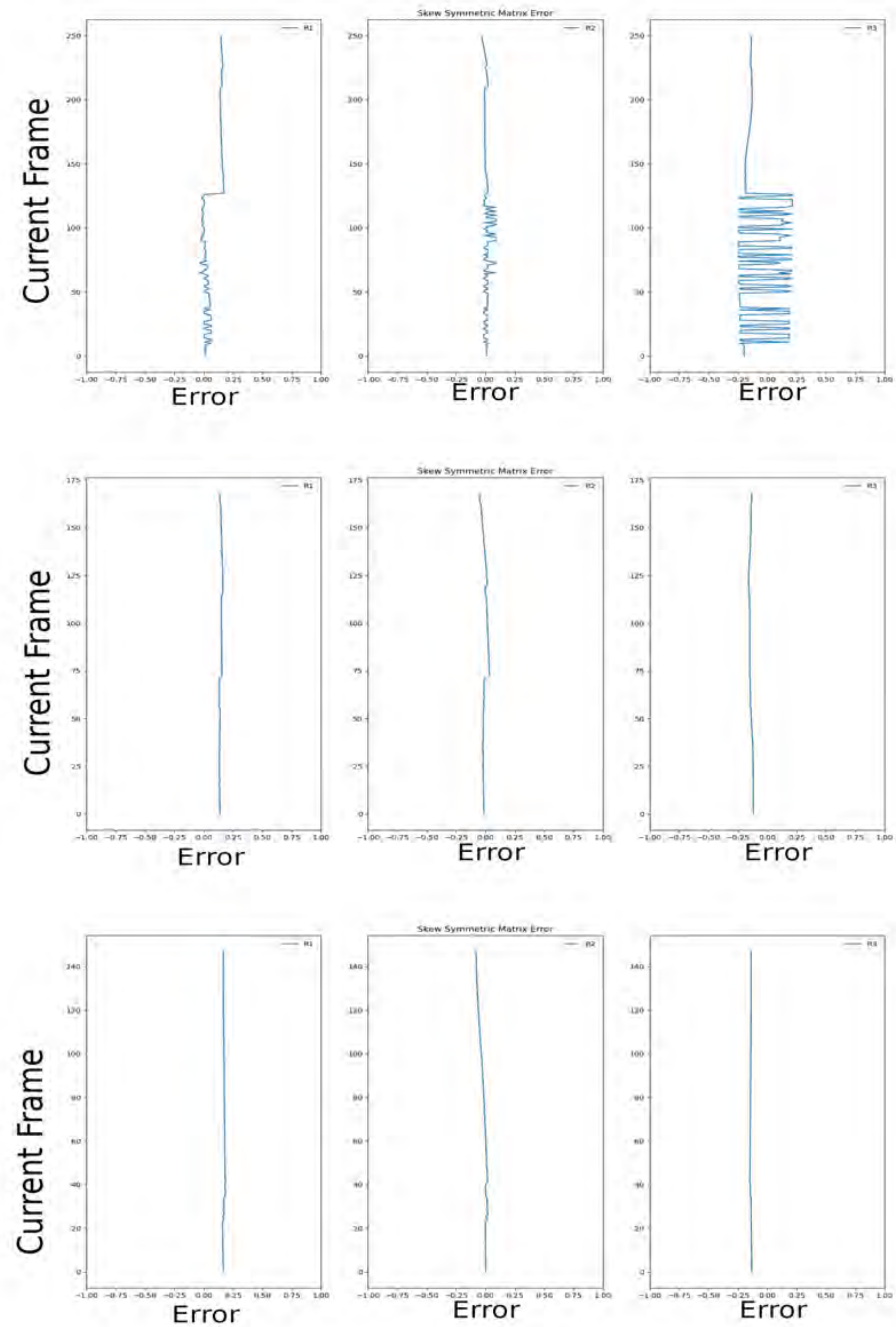

Figure 34: Skew Matrix Results for 3m Skydio Trials

As the UAV performs trials higher in the air the skew error decreases as seen by observing the trial data included. This decrease in error is due to the number of markers being detected in each frame and included in the PNP algorithm increases when the UAV performs the inspections higher in the chamber. This is further displayed in the results found by calculating the difference between both the PNP translation, ${}^{v}t_{c}$, and individual translation,${}^{v}t_{c}$, and plotting the information. The PNP method does not find a smoothed solution, which would use pose estimation from previous frames, which explains values spiking to the left and right in this figure.

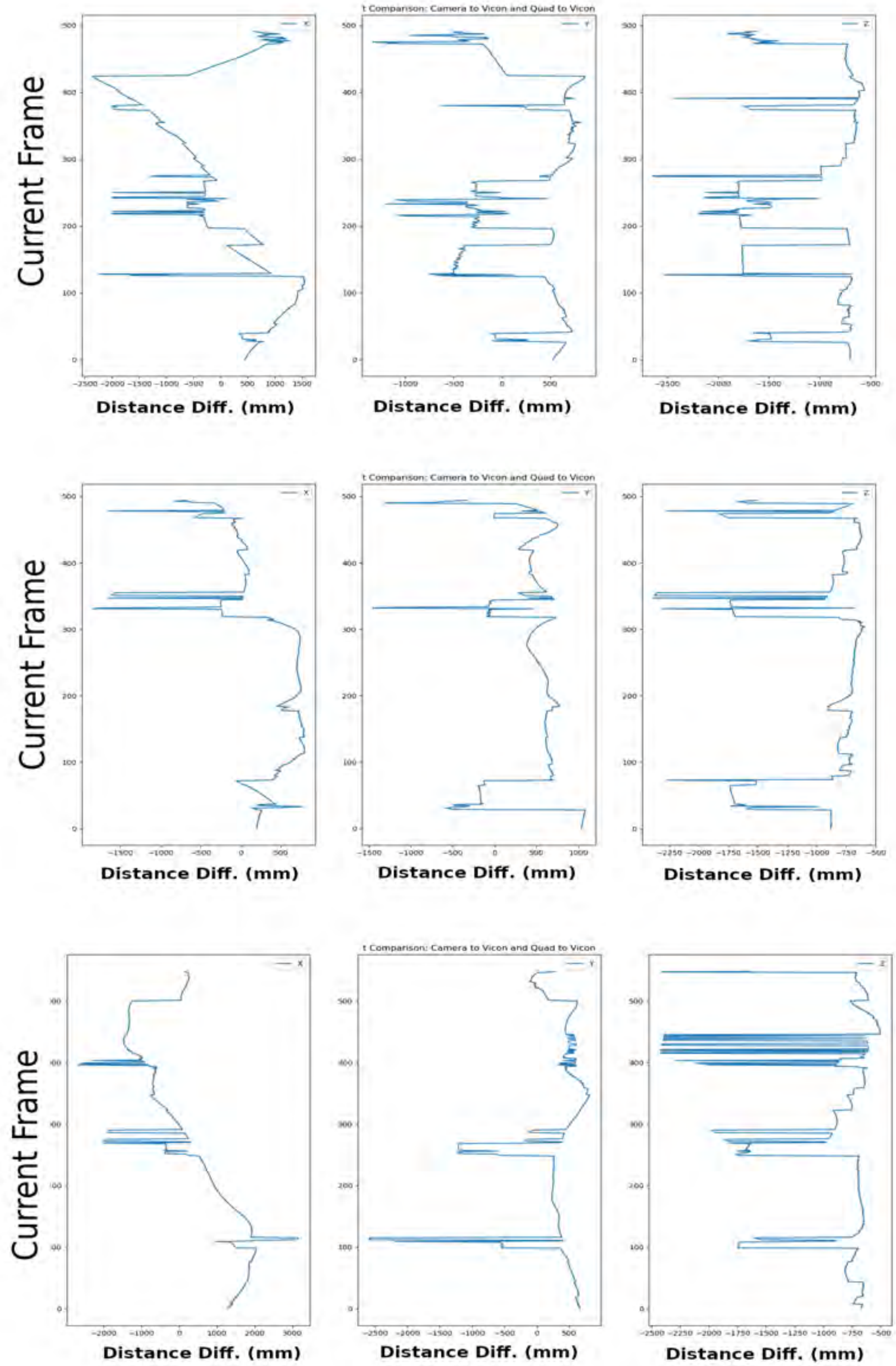

Figure 35: Translation Camera to Vicon (PnP) and Translation Quad to Vicon (est-Pose) Difference: 1m

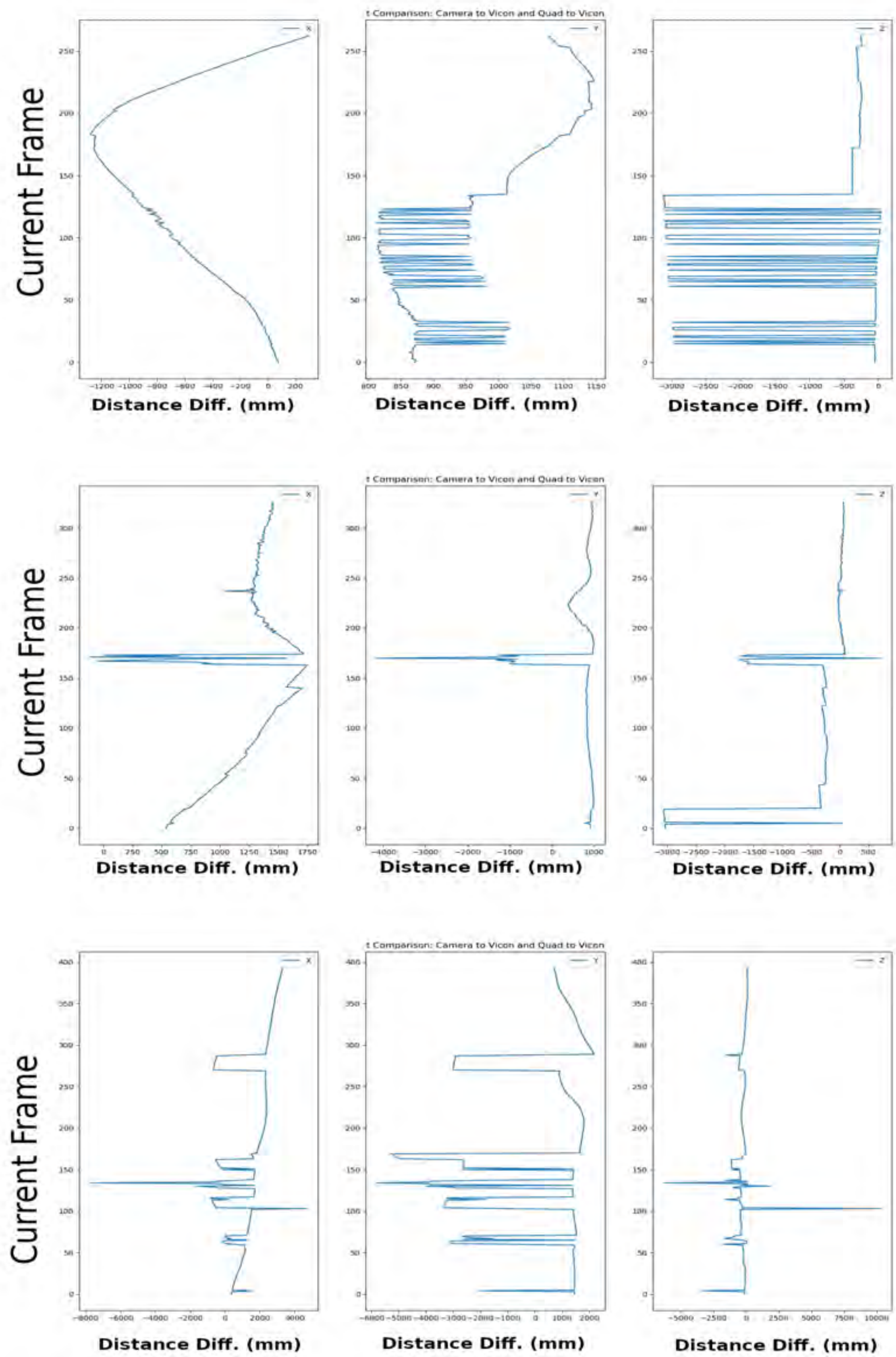

Figure 36: Translation Camera to Vicon (PnP) and Translation Quad to Vicon (est-Pose) Difference: 2m

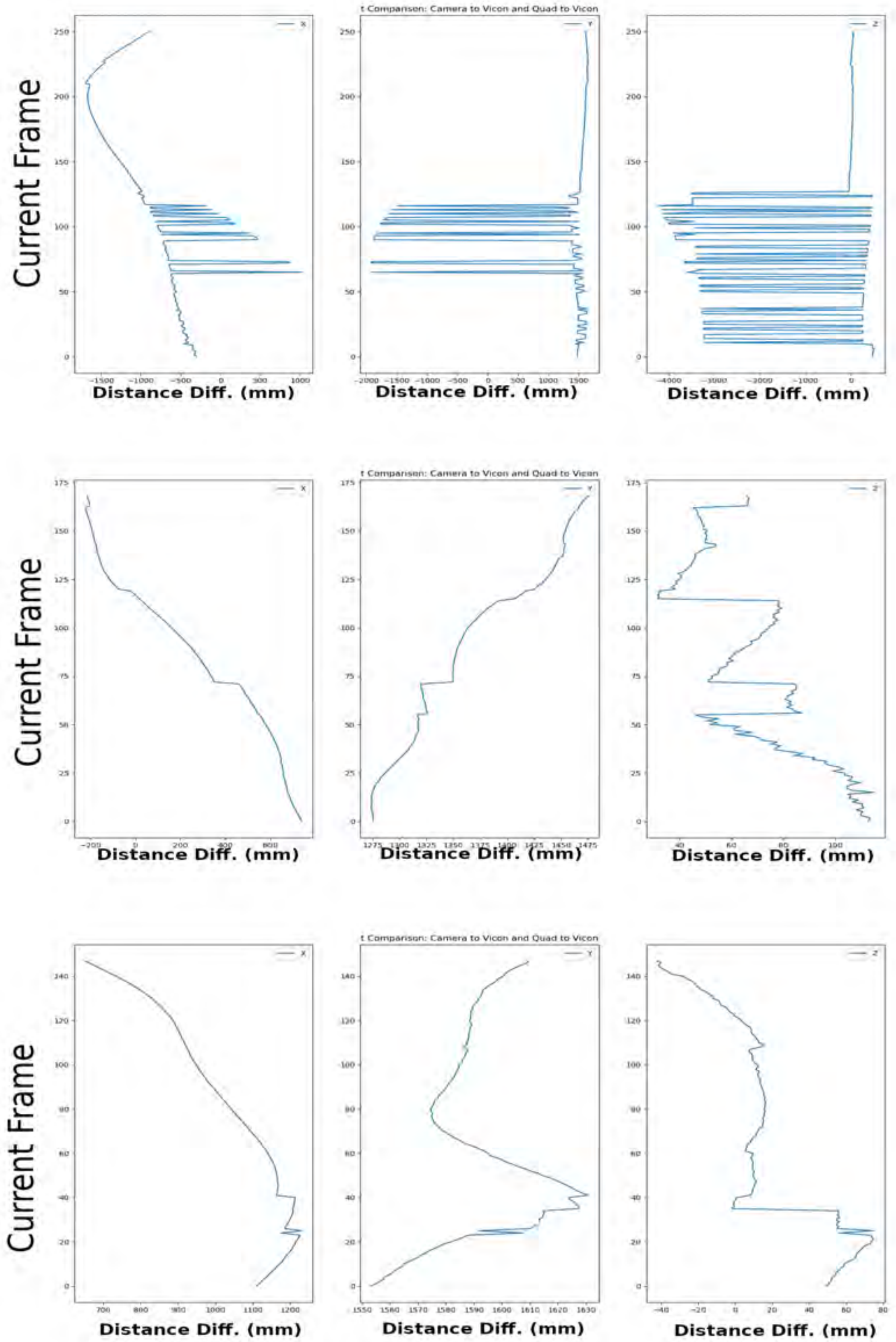

Figure 37: Translation Camera to Vicon (PnP) and Translation Quad to Vicon (est-Pose) Difference: 3m

4.2 Surface Flaw Detection

This work flew trials within a hangar that provided an environment similar to the end use-case of a UAV aircraft inspection system. Imagery from the flights within this hangar over the F-16 show the potential effectiveness of an inspection

Figure 38: F-16 Surface Flaws On Leading Edge

Figure 39: F-16 Surface Flaws On Leading Edge Zoomed-In

Figure 40: F-16 Surface Flaws On Leading Edge Ground View

This image and the series of trials performed with both the Skydio 2 and the Hexrotor within the USAFM Storage Hangar captured imperfections on the surface of the aircraft inspected. Rivets can be counted from the max trial height of 3m as well as oil stains, scratches in paint, and man-made flaws created with painter's tape (representative of exposing primer underneath aircraft paint). Tape was placed along the leading edge of the wing of the aircraft as targets for detection. These man-made flaws can be seen in Figure 38 and more clearly in Figure 39. This footage could be fed to a machine learning algorithm to potentially teach a system to automatically detect surface flaws, which is the end goal desired by the sponsor. It is clear that a 4k capable camera sensor provides ample pixel density to provide the data capturing component for this end-goal system even from a height of 3m.

4.3 Analysis of Alternatives: UAVs

Two UAVs were used in this research as a way to determine the effectiveness of a Commercial off-the-shelf (COTS) system over a built UAV and to explore a variety of on-market capabilities. The use of both vehicles will help future researchers in this area determine which, if either, system is suitable to their research.

4.3.0.1 Sensors

The Skydio 2 includes a 4k 60 Frames Per Second (FPS) capable primary sensor that is capable of taking high resolution photos as well as video. These options were easy to toggle through and select but were mostly predetermined settings that were typically selected before trial flights. These options can be changed in flight with a few taps on the phone that is actively connected to the vehicle which allows the user to easily select FPS, resolution, still/video capture, as well as time-lapse options. This could be useful if manually flying these inspections so the inspection team can actively take photos of damage that they see displayed through the streamed video, though this would not be necessary to implement in a fully automated inspection system. This main sensor was also attached via a gimbal but was set to be in a 90 degree facing down position during the trials. The Skydio 2 always included 6 navigation cameras that are capable of recording flight footage but this data is at present inaccessible to the user. This could provide a more thorough inspection to a maintenance team but would have to be accessed after developing a project with the creators of the Skydio. The Skydio 2 also has Inertial Measurement Unit (IMU)s and Global Positioning System (GPS) capabilities that are included to determine its position during flight, this data is also largely unavailable but could be of great use for future expansion in this area of research.

The Hexrotor was modified to include a 4k 23 FPS camera to complete these trials. This main sensor provided lower frames per second but the resolution was identical to the Skydio 2. The key difference comes with the calibration of this system compared to the automatic calibration that the Skydio 2 performs on boot. The Skydio 2's camera was automatically calibrated and adjusted, but the data from the Hexrotor would need to be calibrated after each change of the lens position. This was only performed once and the lens remained unchanged after. The main sensor of the Hexrotor was mounted at a 90 degree angle but could be altered to be attached on a gimbal easily. This system has GPS and IMUs similarly to the Skydio 2; however, all data was accessible and telemetered to the Ground Control Station (GCS) during the trials. The Hexrotor has an advantage here with the ease of altering and adapting the system to the needs of the user unlike the Skydio 2 which would require the Skydio company's assistance in making any software or hardware changes.

4.3.0.2 Ease of Use

The Skydio 2 was able to be used and could have performed the inspection trials out of the box. The simplicity of its design, combined with its advanced stabilization, imaging, and collision-avoidance capabilities make this a very solid choice when performing manually flown inspections. The system is very safe around expensive assets due to its small weight and size and its always-on collision-avoidance.

The Hexrotor took months to modify to achieve the goals of this research. One of its greatest strengths is the openness to modification and its ability to add additional sensors and batteries to support the changes, but creating a safe, operable system with all sensors working in tandem takes time and a great deal of testing. The Hexrotor is also much larger and heavier which necessitates a well-trained and experienced operator as the pilot of the vehicle. This could be circumvented by using path-planning algorithms to automate the flight of the system but could increase the risk to any expensive assets that are flown around for inspection.

4.3.0.3 UAV Selection

For the current needs of the sponsors this research suggests that a Skydio 2 would perform the majority of the tasks desired. The Skydio 2 would provide ample imagery of any aircraft surface from within a hangar or on a flight line, weather permitting. This system is relatively cheap to procure with a price of under $ 2,000 and it is manufactured and created by a US-based company for purchase and security concerns. This UAV includes collision-avoidance to safe-guard expensive planes that would be inspected. Current maintainers could be taught on the operating requirements of this system in a hangar within a few hours and could perform surface inspections within minutes while detecting trouble spots while the system is still in flight. The desired capabilities that are not provided by this COTS system would be automated surface flaw detection and automated flight path planning.

V. Conclusions

This chapter includes the summary of the work performed during this research. The outcomes of the experiments are recapitulated and the key outcomes are highlighted. A final discussion is included on suggestions for future research and areas to explore following these experiments.

5.1 Summary

The goal of this thesis was to create an algorithm to detect the position of a UAV performing a surface inspection of an aircraft. In the pursuit of this capability, experiments were performed with two different UAVs that recorded the surface of a stationary model aircraft as well as a full-scale aircraft in a series of trials. All trials were performed at 1m, 2m, and 3m heights above the aircraft by two different UAVs. Each UAV was selected for the beneficial traits each platform provided to this research. The Skydio 2 was selected due to its COTS nature and its built-in collision avoidance and recording features. The Hexrotor was selected due to the open and flexible nature of the platform. Due to the Hexrotor being an ANT Center built vehicle it proved to be much more open for data collection for additional metrics such as battery life, GPS position, timing, and general Pixhawk data that could be telemetered in real time.

After these test flights were performed the recorded footage was processed using python and OpenCV libraries. These libraries detected the Euler angles and translation vectors of each ArUco marker found in a recorded frame and calculated the estimated position of the UAV relative to position of the ArUco tags on the floor around the scanned aircraft. After determining the estimated position of the UAV based on the flight videos and the known position of the ArUco tags, the extrinsics for the camera were defined as calculated assumptions. With these data points, the UAV position was calculated for each frame and compared to the Vicon chamber UAV position data.

The individual marker pose estimation method alone was found to be unsuitable for calculating UAV position utilizing ArUco tags alone. A variety of flaws within the testing procedures or with the OpenCV libraries could be the cause of the positional inaccuracy using the ArUco tags. Due to the positioning of the ArUco tags around the stationary aircraft being inspected there are typically only one or two markers detectable in each individual frame. If a larger amount of tags were used to provide increased opportunities for detection the position could have been detected more effectively. The alternative pose estimation method via solvePnPGeneric provided a much more accurate position estimation and was effective at determining the location of the UAV relative to all markers within a captured frame. PNP should be explored further as a pose estimating solution to accurately ascertain the UAV position during flight.

5.2 Future Work

There are many ways that this research could be expanded on or explored in different directions. A few examples of these are listed below:

- **Fiducial Marker Alternatives**: There are a variety of fiducial markers that are supported by OpenCV. These alternative markers could be tested in order to find markers that are the easiest to detect at varying heights like this use case needs. This research could provide more accurate position data which would enable position from markers to be a viable location determination technique for the inspection UAV.

- **Replacement of Marker Dependency**: Other research has explored using landmarking to determine aircraft position. A future researcher could remove the need of fiducial markers entirely if they opted for using a landmarking technique. This could greatly benefit this research area as the end use case for this system would be within an operational hangar. If the hangar would be unable or unwilling to use fiducial markers in the space provided, the UAV could find its position by detecting known symbols on the exterior of the inspected aircraft and estimating the UAV position relative to those pre-determined landmarks.

- **Real-time Experimentation**: The current techniques employed during the experimental phase of this work included receiving some telemetered data from the Hexrotor in flight. The data received included the timing information and Pixhawk data. The ability to send start and end commands for all the dockerized software was also included to enable easier experimentation. A future researcher could extended upon these real-time capabilities by including real-time video transmission to the ground control station and perform the marker detection and position estimation during flight. The operating code for the Hexrotor could be altered to enable this real-time estimation and could lead to a system nearer to the sponsor desired end capability.

- **Inclusion of Other Works**: At the time this research was conducted, two other students were researching other topics related to UAV aircraft surface inspection. These students pursued the detection of the flaws on the skin of the aircraft and a method to determine the position of the flaws within the flight inspection footage. A future researcher could attempt to combine the results and methods utilized across these works to develop a system including all of these capabilities to determine the effectiveness of the system in real-world applications.

5.3 Final Remarks

This researched was a sponsored work exploring UAV-based aircraft surface inspection. The sponsor desires a system that accurately detects flaws on the surface skin of a operational military aircraft that would aid or replace the current human based inspection methods employed by the United States Air Force. This work attempted to solve one component of the automated portion of that desired capability, the ability to locate the position of the UAV performing the inspection. The PNP pose estimation method was able to accurately determine the UAV position, which could be used to determine the location of the aircraft surface flaws. During the experimentation phase of this work the Skydio 2 aircraft was found to meet a large amount of the sponsor's needs. The Skydio 2 is incredibly easy to operate, maintain, and is very low cost. These factors along with its 4K camera and collision avoidance, enabling it to be flown safely near expensive aircraft, could be a solution for the sponsor until a fully developed and custom-built system is created for their use case. The sponsor desired a system that could automatically detect all the surface flaws on an aircraft while decreasing the inspection time from two hours to under three minutes. The Skydio 2 is not capable of flaw detection but would eliminate the need for human inspectors to physical maneuver on and over the aircraft surface which is very time consuming and potentially hazardous. Maintainers could be trained instead to fly this low cost system over the aircraft and the video could then be evaluated by these trained aircraft experts to determine locations that need closer inspection or repair. In the near-term this would provide a large amount of the sponsor's desired capabilities while waiting for a fully developed automatic system to replace it.

Bibliography

1. OpenCV. Perspective-n-point (pnp) pose computation. *OpenCV*.

2. Umberto Papa and Salvatore Ponte. Preliminary Design of an Unmanned Aircraft System for Aircraft General Visual Inspection. *Electronics*, 7(12), dec 2018.

3. Vasileios Tzitzilonis, Konstantinos Malandrakis, Luca Zanotti Fragonara, Jose Angel Gonzalez Domingo, Nicolas P. Avdelidis, Antonios Tsourdos, and Kevin Forster. Inspection of Aircraft Wing Panels Using Unmanned Aerial Vehicles. *Sensors*, 19(8), apr 2019.

4. Changhong Fu, Adrian Carrio, Miguel A. Olivares-Mendez, Ramon Suarez-Fernandez, and Pascual Campoy. Robust real-time vision-based aircraft tracking from Unmanned Aerial Vehicles. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2014.

5. Rui Filipe Fernandez, Kevin Keller, and Jeffrey Robins. Design of a system for Aircraft Fuselage Inspection. In *2016 IEEE Systems and Information Engineering Design Symposium (SIEDS)*. IEEE, apr 2016.

6. Yasmina Benkhoui, Tahar El Korchi, and Ludwig Reinhold. UAS-Based Crack Detection Using Stereo Cameras: a Comparative Study. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, jun 2019.

7. Patrick Silberberg. Aircraft Inspection by Multirotor UAV Using Coverage Path Planning. 2021.

8. Martin Bugaj, Andrej Novak, Anna Stelmach, and Tomasz Lusiak. Unmanned Aerial Vehicles and Their Use for Aircraft Inspection. In *2020 New Trends in Civil Aviation (NTCA)*. IEEE, nov 2020.

9. Dario Cazzato, Miguel A. Olivares-Mendez, Jose Luis Sanchez-Lopez, and Holger Voos. Vision-Based Aircraft Pose Estimation for UAVs Autonomous Inspection without Fiducial Markers. In *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, oct 2019.

10. Gennaro Ariante, Umberto Papa, Salvatore Ponte, and Giuseppe Del Core. Uas for positioning and field mapping using lidar and imu sensors data: Kalman filtering and integration. In *2019 IEEE 5th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, pages 522–527, 2019.

11. Julien Miranda., Stanislas Larnier., Ariane Herbulot., and Michel Devy. Uav-based inspection of airplane exterior screws with computer vision. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 4: VISAPP,*, pages 421–427. INSTICC, SciTePress, 2019.

12. Sophie Jordan, Julian Moore, Sierra Hovet, John Box, Jason Perry, Kevin Kirsche, Dexter Lewis, and Zion Tsz Ho Tse. State-of-the-art technologies for uav inspections. *IET Radar, Sonar and Navigation*, 12:151–164, 2 2018.

13. FAA. Airplane Flying Handbook (FAA-H-8083-3B) Chapter 2. 2021.

14. Monroe Aerospace. How Often Do Commercial Airplanes Need Maintenance? Oct 2020.

15. Md. Alahi Uddin Khan. Non-destructive Testing Applications in Commercial Aircraft Maintenance. 1999.

16. CJ Alberts, CW Carroll, and WM Kaufman. Automated Inspection of Aircraft. (April), 1998.

17. Boyong He, Bo Huang, Yuxing Lin, and Liaoni Wu. Intelligent unmanned aerial vehicle (UAV) system for aircraft surface inspection. In *2020 7th International Forum on Electrical Engineering and Automation (IFEEA)*. IEEE, sep 2020.

18. Edmundo Guerra, Rodrigo Munguía, and Antoni Grau. Uav visual and laser sensors fusion for detection and positioning in industrial applications. *Sensors*, 18(7), 2018.

19. Guohao Zhang and Li-Ta Hsu. Intelligent gnss/ins integrated navigation system for a commercial uav flight control system. *Aerospace Science and Technology*, 80:368–380, 2018.

20. Slawomir Grzonka, Giorgio Grisetti, and Wolfram Burgard. Towards a navigation system for autonomous indoor flying. pages 2878–2883, 2009.

21. A. : Košt'ák, M.; Slabý. Designing a Simple Fiducial Marker for Localization in Spatial Scenes Using Neural Networks. 2021,21, 5407.

22. S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. 2014.

23. R. C. Fischler, M. A.; Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. 1981.

24. Bernhard Thalheim Hannu Jaakkola. Architecture-driven modelling methodologies. pages 97 – 116, 2011.

25. Lorraine Brisacier-Porchon, Omar Hammami Ensta, and Rémi Boutemy. Modeling a uav in practice: A comparison between rhapsody and capella. pages 1–8, 2021.

26. Nadia A. Tepper. Exploring the use of model-based systems engineering (mbse) to develop systems architectures in naval ship design. 2010.

27. Kaitlin Henderson and Alejandro Salado. Value and benefits of model-based systems engineering (mbse): Evidence from the literature. *Systems Engineering*, 24(1):51–66, 2021.

28. Addison Sears-Collins. How to perform pose estimation using an aruco marker. *Automatic Addison*, 2021.

29. Caleb B. Schmidt. Aircraft surface inspection scripts opencv/python. 2021-2022.

Acronyms

AFIT Air Force Institute of Technology. 2, 5

AFRL Air Force Research Laboratory. 3, 5

ANT Autonomy and Navigation Technology. 5

CATIA Computer Aided Three-Dimensional Interactive Application. 12

COTS Commercial off-the-shelf. 2, 13, 67

CPU Central Processing Unit. 15

DoD Department of Defense. 1

FPS Frames Per Second. v, 14, 18, 30, 67

GCS Ground Control Station. 23, 28, 40, 68

GPS Global Positioning System. 67

GPU Graphics Processing Unit. 15

GVI General Visual Inspection. 8

IBD Internal Block Diagram. 12, 19, 28

IMU Inertial Measurement Unit. 6, 67

IR Infrared. 15

LCM Lightweight Communications and Marshalling. 34, 49

LIDAR Light Detection and Ranging. 6

MBSE Model-Based Systems Engineering. 12

NDT Non-destructive testing. 8

PnP Perspective N- points Problem. 10, 46

PTP Precise Time Protocol. 34

PTZ Pan Tilt Zoom. 7

RCS Radar cross-section. 8

SOC System On Chip. 15

SSD Solid State Drive. 34

SUAS Small Unmanned Aerial System. 12

UAS Unmanned Aerial System. 1

UAV Unmanned Aerial Vehicle. 2, 50

USAF United States Air Force. 1, 2

USAFM United States Air Force Museum. 2, 35, 50, 66

WPAFB Wright-Patterson Air Force Base. 34

ZoI Zones of Interest. 7

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*
18–03–2022

2. REPORT TYPE
Master's Thesis

3. DATES COVERED *(From — To)*
Sept 20 — Mar 2022

4. TITLE AND SUBTITLE
UAV POSITIONING DATA DETERMINED VIA ARUCO TAGS FOR AIRCRAFT SURFACE INSPECTION

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Schmidt, Caleb, Capt, USSF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT-ENG-MS-22-M-059
AFIT-ENV-MS-22-M-258

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFRL/RX
Materials State Awareness Supportability Branch Building 653
WPAFB OH 45433-7765

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Aircraft are frequently inspected to ensure that military and civilian safety standards are adhered to. These inspections are performed pre- and post-flight and are currently performed by trained maintenance personnel. This work furthers the automation of aircraft surface inspection by using ArUco tags to determine the position of the UAV during aerial inspections. The ArUco tag based position data was then compared to a highly accurate infrared motion capture system to determine the viability of this for accurate positioning of the vehicle. This work includes flight experiments with two different UAVs to perform a system viability comparison.

15. SUBJECT TERMS
unmanned aerial vehicles (UAV), Inspection, unmanned aerial systems (UAS), small unmanned aerial systems (sUAS), Aircraft, Vision-based, Positioning, ArUco, OpenCV

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
					Dr. Robert Leishman, AFIT/ENG
					19b. TELEPHONE NUMBER *(include area code)*
U	U	U	UU	91	(312) 785-3636 x4755; robert.leishman@afit.edu