DREXEL UNIVERSITY
Graduate College

Thesis Approval Form

Title: Structural Health Monitoring using Unmanned Aerial Systems

Author: Andrew James Ellenberg

This thesis is hereby accepted and approved.

Signatures:

Examining Committee:

Chair:

Dr. Antonios Kontsos

Members:

Dr. Ivan Bartoli

Dr. B. C. Chang

Dr. Ani Hsieh

Dr. Matthew McCarthy

Academic Advisor:

Diane Venti

Department Head:

Dr. David Miller

Graduate College 3141 Chestnut Street Main 301 Philadelphia, PA 19104

Tel: 215.895.0366 Fax: 215.895.0495

Email: graduatecollege@drexel.edu Web: www.drexel.edu/graduatecollege

Structural Health Monitoring using Unmanned Aerial Systems

A Thesis

Submitted to the Faculty

of

Drexel University

by

Andrew James Ellenberg

in partial fulfillment of the

requirements for the degree

of

Doctor of Philosophy

in

Mechanical Engineering and Mechanics

April 2017

Dedications

I dedicate this thesis to my mother and father, Kathy and Paul Ellenberg, and my sister, Karly Redrow, who have been so supportive and helped me so much throughout my life and helped me to achieve this Doctorate Degree. Mom and Dad, I have always looked up to you and you have always set a great example for me. You are the most amazing parents and there is no possible way for me to express in words how grateful I am for everything you have done for me and how much I love you. I am who I am because of you and I will never be able to thank you enough for that. Karly, thank you for being the best sister anyone has ever had and I love you so much!

Drew

Acknowledgments

I am very thankful to everyone who has supported me throughout my career at Drexel University. Most importantly, I would like to thank Dr. Kontsos and Dr. Bartoli for convincing me to continue my education and supporting me for the past five years through my senior design project and the Ph.D. program. They have been my teachers, my mentors, my bosses when needed, but most importantly my friends and there are not enough words to express how grateful I am for everything they have done to help me succeed. They stood by me through good times and bad and I would not have been able to do this without them.

I would also like to thank all of the friends who have supported me. Lara Angelone and Alison Krick were the two best senior design partners I could have ever asked for. They are the reason I met Dr. Kontsos and Dr. Bartoli and they helped me make my decision to go to graduate school. I also want to thank the current and past members of the Theoretical and Applied Mechanics Group have helped me with support for experiments, collaboration on projects and the many great times we have had outside of research. Dr. Kavan Hazeli, Dr. Prashanth Vanniaparambil, Dr. Jefferson Cuadra, Mr. Shane Esola, and Dr. Konstantinos Baxevanakis have helped me so much during senior design and throughout my graduate career. They have been a great example for me to follow, always encouraged me to do good work, answered my many questions, and gave me advice that has helped me in my research and in my life. I also want to thank Dr. Franklin Moon, Dr. Emin Aktan, Dr. Matteo Mazzotti, Nestor Casteneda, Mike Cabal, Aaron Goldberg, Graceson Kalathil, Brian Wisner, Melvin Mathew, Shi Ye, Qiang Mao, Mustafa Furkan, Fuad Khan, John DeVitis, John Braley, David Masceri, Nick Ramano, Michelle

Burnworth, and Charles Young for all of their support during lab and field experiments, advice, and wonderful conversations we have had during my time at Drexel.

I would also like to thank Piasecki Aircraft Corporation, specifically Fred Piasecki, Brian Geiger, and Chris Messineo for all of their help and expertise in helping us build an unmanned aerial system. Thank you for the many hours and weekends you spent helping myself and the senior design team build the system as well as all of your support throughout my PhD. I would also like to thank Jason Allen, Harold Donovan, Maria Tabbut, Luis Castro, and John Swearer who were in the senior design team that built the full system with the help of Piasecki Aircraft Corporation and taught me so much. I would like to acknowledge Trilion Quality Systems for the use of the TRITOP system, Dr. Ani Hsieh and Dr. Frank Moon for allowing me to use their laboratory facilities at Drexel and for providing feedback at various stages of my work as well as Mr. Matthew Nial and Mr. Eric Cristofalo for their assistance in the development of the algorithms during my senior design project. Finally, I would like to acknowledge the American Society for Nondestructive Testing, the National Science Foundation and the Federal Highway Administration for funding this research. I also want to thank the Pennsylvania Department of Transportation, Rutgers University, the Burlington County Bridge Commission, and the West Virginia Department of Transportation for their support during my time at Drexel University.

Table of Contents

List of Tables

List of Figures

Abstract

Structural Health Monitoring using Unmanned Aerial Systems

Andrew James Ellenberg

The use of Structural Health Monitoring (SHM) techniques is paramount to the safety and longevity of the structures. Many fields use this approach to monitor the performance of a system through time to determine the proper time and funds associated with repair and replacement. The monitoring of these systems includes nondestructive testing techniques (NDT), sensors permanently installed on the structure, and can also rely heavily on visual inspection. Visual inspection is widely used due to the level of trust owners have in the inspection personnel, however it is time consuming, expensive, and relies heavily on the experience of the inspectors. It is for these reasons that rapid data acquisition platforms must be developed using remote sensing systems to collect, process, and display data to decision makers quickly to make well informed decisions based on quantitative data or provide information for further inspection with a contact technique for targeted inspection. The proposed multirotor Unmanned Aerial System (UAS) platform carries a multispectral imaging payload to collect data and serve as another tool in the SHM cycle. Several demonstrations were performed in a laboratory setting using UAS acquired imagery for identification and measurement of structures. Outdoor validation was completed using a simulated bridge deck and ground based setups on in service structures. Finally, static laboratory measurements were obtained using multispectral patterns to obtain multiscale deformation measurements that will be required for use on a UAS. The

novel multiscale, multispectral image analysis using UAS acquired imagery demonstrates the value of the remote sensing system as a nondestructive testing platform and tool for SHM.

This page is intentionally left blank

Chapter 1 Introduction

1.1 Motivation

The 2008 (USGAO-08-763T) testimony to Congress stated that the United States economy is "dependent on the reliability, safety, and security of its physical infrastructure" and therefore it is necessary to preserve the health of these complex systems [1]. This task has become increasingly challenging due to the age, scale, and quantity of the structures. The extent of United States infrastructure has grown due to higher demands and now there are over 610,000 bridges, over 5,500,000 commercial buildings, over 160,000 miles of railroad tracks, 4,000,000 miles of roads, 84,000 dams, over 19,000 airports, and 400,000 miles of electric transmission lines [2, 3]. The American Society of Civil Engineers (ASCE) 2017 infrastructure report card scores the overall United States infrastructure at a D+ [4]. An estimated 3.6 trillion dollar investment will be required through 2020 to improve the state of infrastructure to meet the increasing demands [2]. Even if this money is invested, another major challenge is to determine how to prioritize which structure is repaired with the funds. There are many different ways to prioritize the maintenance and repair, including the importance to industry, importance to decision makers, or the extent of the damage. In addition, it is important to determine what is considered to be damaged enough to require repair to ensure that money is not wasted. Structural health monitoring (SHM) is one way to help determine the answers to these questions to repair or replace the right structure at the optimal time with the best method [5].

Using bridges as a representative example, Figure 1 shows the number of structurally deficient bridges, rated 4 and below on a scale of 0 to 9, in the United States.

Bridges were assessed with a grade of C+, which is better than the average grade of United States infrastructure, and yet still, major problems exist in many parts of the country, particularly in the Northeast and northern parts of the Midwest [4, 6]. Infrastructure inspection is expensive due to the time and frequency of the inspections resulting from the scale and quantity of the structures. With this in mind, the more money that is spent on inspection, the less money there is available for repairs. This forces rate of inspections down, for instance, bridges must be inspected at least once every two years and the 9.1% of bridges deemed structurally deficient must be inspected more frequently. This means that every day, over 800 bridges must be inspection to keep up with this rate [7]. Structurally deficient bridges require a higher degree of monitoring due to heavily damaged or degraded areas. At a rate of an inspection once every two years, it is a very discrete form of SHM.

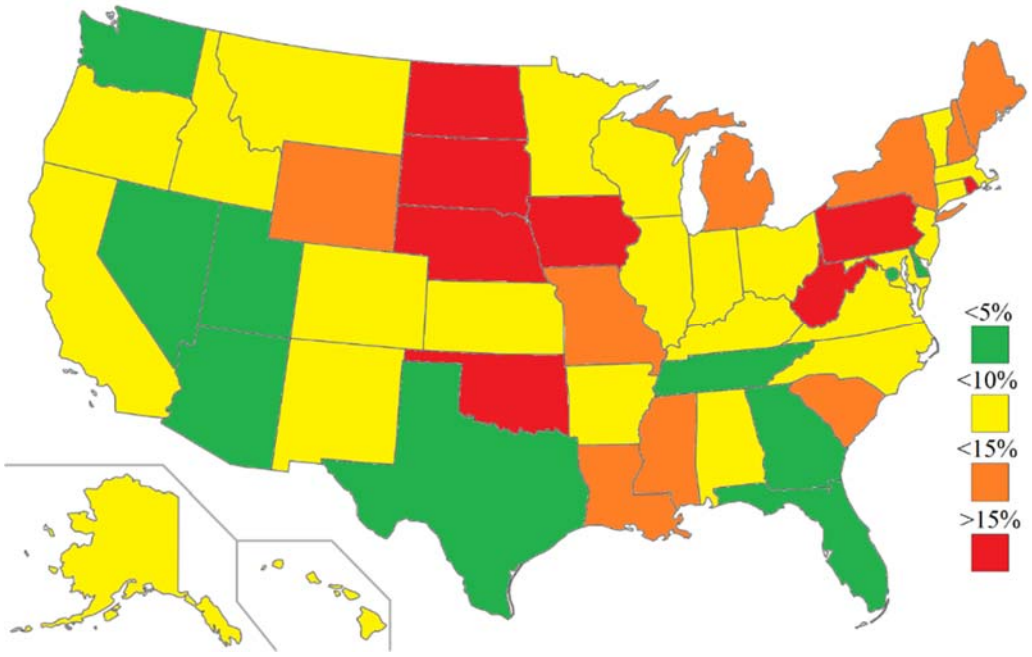


Figure 1. Map of the United States with color codes corresponding to percentage of structurally deficient bridges [6, 7]

America's challenge today is to meet the growing demand of its aging infrastructure by rebuilding, maintaining, modernizing, and expanding its infrastructure while attempting to limit the costs associated with assessment, maintenance, and repair. Figure 1 shows a map of the United States color coded with the corresponding percentage of structurally deficient bridges by state according to data from 2016 [6, 7].

Visual inspection is the most heavily relied upon method of inspection because it has historically been the most intuitive and effective method for evaluation. However, it is often time consuming, expensive, produces qualitative results, and relies heavily on the experience of the inspection personnel [8-10]. Visual inspection also requires adequate access to remote locations, which can increase the associated personal safety risk as well as add costs for equipment such as scaffolding, lifts and other protective equipment. These inherent limitations, associated with high stress situations that are sometimes involved in visual inspection, can hamper rapid and quantitative-based decisions to be made regarding necessary repairs [11].

In order to supplement the qualitative results of visual inspection, structural health monitoring (SHM) and nondestructive testing (NDT) techniques are being utilized more frequently on many structures to contribute more information about the structure [12-14]. Some of these methods include fiber Bragg gratings, strain gauges, accelerometers, temperature sensors, impact echo, eddy current testing, acoustic emission, ultrasonic testing, and many others [13, 15-21]. Obtaining useful information from the data is a challenge both for visual inspection and nondestructive testing techniques because the ultimate goal of the inspection is to make a decision about allocating funding for

maintenance [22]. The safety and reliability of the structure depends heavily on the decisions made after analyzing the data from the inspection.

SHM is defined as a damage detection strategy which is of importance to all structures [23]. One of the well-known, recent failures was the I-35 bridge collapse in Minnesota, which killed 13 people and injured 145 people [24]. Figure 2 (a) shows an image taken in 2003 with gusset plates in the field of view that appear to be bent [24]. The bridge did not collapse until August, 2007 meaning that at least two inspections had occurred without the problem being addressed. Two images of this failure are shown in Figure 2 (b) [24].

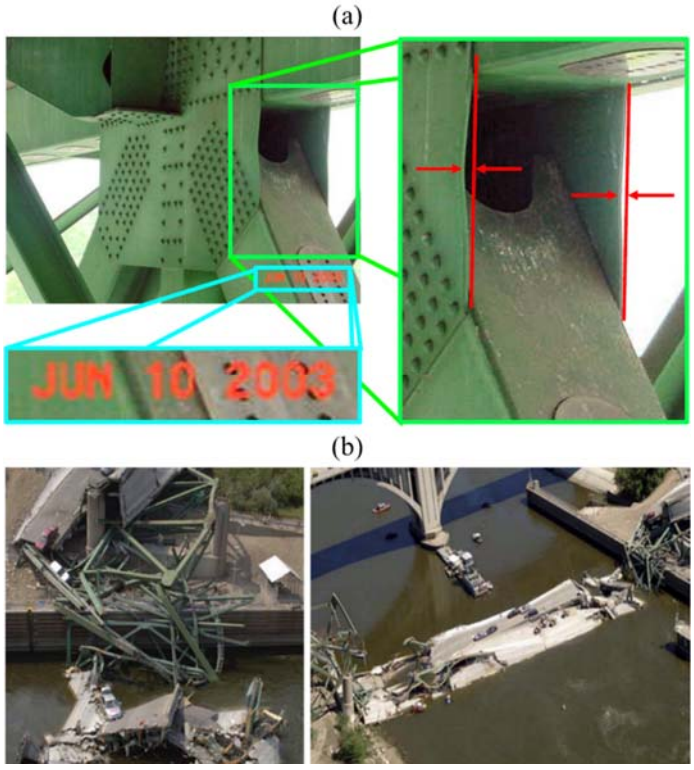


Figure 2. Images of the I-35 bridge collapse [24]

It is for this type of failure that SHM must be implemented because the overarching goal of SHM is to prevent failures. Though this collapse was caused by many different factors including a mistake in steel replacement of a gusset plate years earlier, rush hour traffic, weight added to the structure due to repairs, and a large amount of construction materials adding to the dead load, a closer monitoring of the structure may have been able detect at least one of the problems to prevent this failure [24].

Given the large scale of civil infrastructure systems, the inspection process is hindered by access issues resulting in higher equipment costs and longer time required to perform an inspection, resulting in longer time intervals between inspections due to limited funds. An example of the difficulties associated with access to difficult to reach areas of the structure is shown in Figure 3 where a barge and a lift were required to get the personnel to the area of the structure that required inspection.

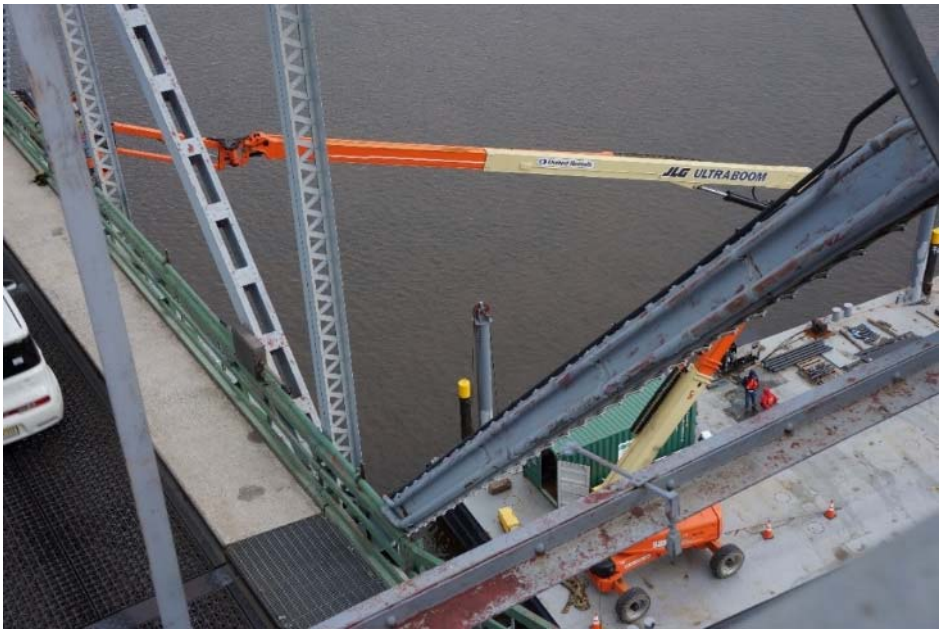


Figure 3. Access difficulties encountered during an inspection.

Low-cost access to remote locations with high resolution images (e.g. RGB, infrared) from repeatable positions by Unmanned Aerial Systems (UAS) will positively impact how infrastructure is assessed and how maintenance is prioritized [25]. For example, these systems could permit tracking deterioration at a much higher temporal resolution, and therefore provide a better understanding of how changes occur with time to improve forecasting models. In order to realize this potential however, the extraction of quantitative information from images is required. Much has been written about the shortcomings of visual inspection procedures (especially related to their high variability and subjective nature [8, 26, 27]), and without the ability to extract quantitative information from images, the promise of UASs will remain limited for similar reasons. To address these deficiencies, attention must be focused on extracting quantitative information from images including the identification of features, such as cracks, corrosion, and other regions of deterioration, and quantification of mechanical responses, such as displacements and strains.

1.2 Research Objectives

UAS have had significant technological growth in commercial applications in the past few years. There have been many studies done demonstrating how UAS can collect data using a camera which could potentially save money compared to traditional methods of access to difficult to reach areas. Several companies have been formed to use UAS to collect data and create maps of the terrain below from the images including Pix4D, PrecisionHawk, Drone Deploy, Skycatch, and many others. In this context, UAS are very useful for collecting data and relying on a user to identify potential areas that need further examination. However, UAS imagery and video can often be disorienting to sort through

so visualization of the data in the form of stitched imagery has been developed. Three dimensional models have been created to better assist in the analysis by a user [28-30]. Most of the work done in this area focuses on collecting and visualizing the data, however, these are just the first steps of what is required to fully see the potential of the UAS for commercial applications.

The research objective of this thesis is to develop and validate techniques for acquiring data from unmanned aerial system remote sensing platform and converting it into useful damage information to aid in the evaluation of infrastructure. This work uses bridge structural health monitoring (SHM) as an example, however it is not limited to bridges. Roadways, dams, powerlines, pipelines, wind turbines, power plants, and many other structures could benefit from using UAS remote sensing for SHM applications. To accomplish this goal, the work presented in this thesis includes automated damage identification and visualization from imagery, measurement of size and deformation, and measurement using multispectral imagery to selectively choose an optimal pattern for analysis using Digital Image Correlation (DIC). This undertaking is crucial to determining the effectiveness of a UAS platform as a future inspection tool to provide decision makers with fast, reliable, and useful information to make more informed decisions.

1.3 Thesis Outline

This thesis addresses some of the challenges associated with UAS data to provide useful information by providing an automated identification of areas of interest and a new way to use UAS for the measurement of deformation using imagery. Chapter 2 presents the background and state-of-the art in the use of UAS for remote sensing applications.

Chapter 3 presents the objectives of this work, hypotheses of the studies, and approach used to accomplish these goals.

Chapter 4 presents the identification of surface damage from imagery and flag them as areas of interest for further analysis. This chapter also demonstrates the use of a global view of a structure. Chapter 5 presents the identification of subsurface damage on a bridge deck using a post processing algorithm. It also presents the usefulness of a global view of the structure with both surface and subsurface damage overlaid on the map of the structure and a way to collect color and infrared data using a UAS to aid in the post processing of the imagery.

Chapter 6 presents the techniques used to obtain measurements of a structure from image data. This chapter presents the use of fiducial markers for analysis as well as full field data to measure deformation in both a static and hovering configuration. A novel technique is presented to use multiscale, multispectral imagery to obtain measurements at different distances from the target structure. Chapter 7 explains some of the future work and extensions of the research presented and the concluding remarks.

Chapter 2 Background and State-of-the-Art

2.1 Structural Health Monitoring (SHM)

Structural health monitoring (SHM) encompasses a variety of techniques and procedures implemented as a damage detection strategy that continuously monitors a structure through time to obtain information about the structure to predict its lifecycle [31]. There are many fields and systems that rely on SHM techniques to keep the systems safe and in general, these applications rely on the axioms of SHM defined by Worden et al. and shown in Table 1 [32].

Table 1. Axioms of SHM [32]

Axiom 1	Flaws and defects are present in all materials.
Axiom 2	At least two states are required to perform an assessment of damage.
Axiom 3	Machine learning is a critical element that can be utilized for SHM. Damage identification and localization can be done with unsupervised learning, but determining the type of damage and extent requires supervised learning.
Axiom 4	Signal processing and feature extraction are required to identify damage because sensors do not measure damage. Intelligent choices of features will determine the sensitivity and performance of the identification of damage.
Axiom 5	The SHM sensing system must be determined based on the spatial and time scales as well as the type of damage.
Axiom 6	All signal processing is dependent on a tradeoff between sensitivity and noise.
Axiom 7	The detection of damage from changes in system dynamics is inversely proportional to the frequency of excitation.

Axiom 1 highlights that not all material defects and flaws are damage because damage is associated with the degradation in performance of the system [23, 32]. The performance however is determined by the application and the extent of damage can be

associated with material defects, or in most cases, be insignificant until the material defects connect to cause a larger change in response of the system. The second axiom makes the point that there has to be a baseline to which the current state is compared to in order to determine if damage has occurred. For example, if a strain gauge is placed after a large amount of damage is present, the measurements will not be useful because the baseline is already damaged. It would therefore be better to place the strain gauge prior to damage initiation and compare the measurements to the undamaged state. Axioms 3 and 4 explain that feature extraction and machine learning are important parts of SHM because an enormous amount of data can be collected making it impossible for users to analyze all of the data manually. In addition, decisions need to be made based on the data and machine learning as well as experience from known previous cases can be leveraged to help aid in these decisions [33]. Furthermore, big data is a challenge in many applications including SHM due to the length of time required for measurement, the speed of data capture, and the number of sensors required for measurements. Axiom 5 explains that the proper choice of a sensor depends on what requires measurement and it is important to use the right sensor for the right application. Sensitivity versus noise is a challenge for all signal processing algorithms and is highlighted in axiom 6. A false alarm and missed detection rate will always be present when applying any detection method, but it is up to the user to determine the acceptable limits of false alarms and missed detections and examine the cost of each to determine if one is more acceptable than the other. The last axiom describes that the length of time between measurements will determine the accuracy of the damage growth monitoring [32].

SHM has been applied to a broad range of fields with slight variations such as aerospace [12] and large scale infrastructure [34]. Many different approaches and sensors are used for different systems, however the goal is always to identify areas of damage quickly for repair to ensure the health of the structure does not degrade to an unacceptable level. In most SHM applications, NonDestructive Testing (NDT) is implemented to provide information related to the structural behavior and the progressive deterioration through time. NDT techniques include impact echo, infrared thermography, ground penetrating radar, eddy current testing, ultrasonic testing, acoustic emission, radiography, and many others [17]. All of these techniques are used to identify different types of damage or flaws in the material making them not directly SHM methods, however if the techniques are applied over time and used to track growth of damage, these methods become tools in the SHM approach.

The overall goal in SHM is to collect data, identify damage, assess the damage, and finally determine corrective action. The axioms presented in Table 1 provide the basic guidelines that are used for SHM and what must be accepted and determined when forming a SHM strategy. Though most applications of SHM are time based, condition based maintenance would be more cost effective and achieve a higher level of safety [23]. A closer monitoring of the systems and a heavy reliance on automated data collection and machine learning will ultimately aid in the prominence of condition based assessment and maintenance.

Among other asset management tools, SHM is essential to keep civil infrastructure safe and functional. The cycle of SHM for bridges is shown below in Figure 4. The first part of the SHM cycle is the collection of data through inspection or sensing systems. The

next step is to analyze the data acquired and finally, the last step is to assess the structure and determine best course of action. Then the process repeats for the duration of the system's lifecycle.

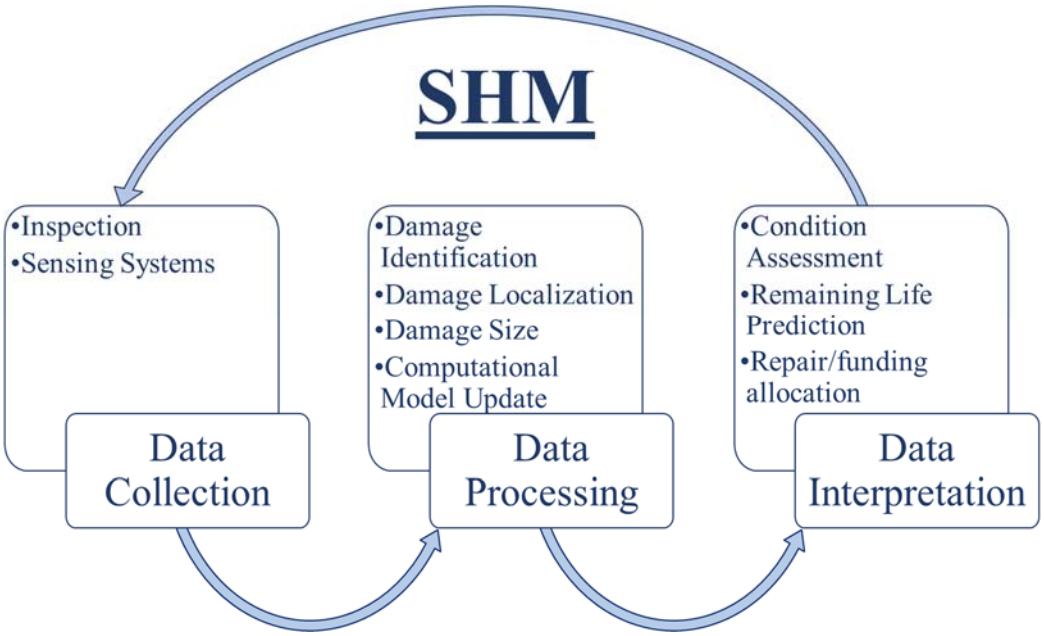


Figure 4. SHM cycle

In this thesis, bridges are often used as a representative example for assessment and SHM. In this context, visual inspection is the primary method of evaluation for the more than 610,000 bridges in the United States. The inspections are time based and must occur at least once every two years. Of these 610,000 bridges, 9.1% of them are structurally deficient and many more are functionally obsolete. Functionally obsolete bridges are bridges that were not designed to accommodate the current demands. Bridges labeled structurally deficient are not unsafe, they do have a bridge condition rating of 4 or less. The bridge condition rating is based on the deck, superstructure, and substructure and is shown below in Table 2 [6]. It is often difficult to determine which rating a bridge should receive

since the metrics to determine which category a bridge should be assessed are qualitative, subjective, and based on the experience of the inspection personnel. It is for this reason that the results for inspections are inconsistent among different inspectors, sometimes ranging two points on the bridge condition rating scale [8].

Table 2. Bridge Condition Rating Scale [6]

Rating	Condition Category	Description*
9	Excellent	
8	Very Good	No problems noted.
7	Good	Some minor problems.
6	Satisfactory	Structural elements show some minor deterioration.
5	Fair	All primary structural elements are sound but may have minor section loss, cracking, spalling, or scour.
4	Poor	Advanced section loss, deterioration, spalling, or scour.
3	Serious	Loss of section, deterioration, spalling, or scour have seriously affected primary structural components. Local failures are possible. Fatigue cracks in steel or shear cracks in concrete may be present.
2	Critical	Advanced deterioration of primary structural elements. Fatigue cracks in steel or shear cracks in concrete may be present or scour may have removed substructure support. Unless closely monitored, it may be necessary to close the bridge until corrective action is taken.
1	Imminent Failure	Major deterioration or section loss present in critical structural components, or obvious loss present in critical structural components, or obvious vertical or horizontal movement affecting structural stability. Bridge is closed to traffic, but corrective action may be sufficient to put the bridge back in light service.
0	Failed	Bridge is out of service and is beyond corrective action.

Structurally deficient bridges must be monitored more closely to ensure that the damaged areas to not cause the bridge to close. The field process of SHM is shown in

Figure 5. Inspection is a large part of the current field practice of SHM for bridges and it is by far the most trusted and accepted form of assessment for bridges followed by strain gauges. The data processing is often qualitative and determined by the inspection personnel. The results of the data processing are then submitted to the bridge owner and a decision is made to determine the best action. The biggest gap in the SHM cycle is the data processing step which takes the data to information regarding the health of the structure. The large time span between inspections makes it a very discrete form of SHM. Therefore, the final segment connecting the condition assessment to the next inspection is not shown in the figure.

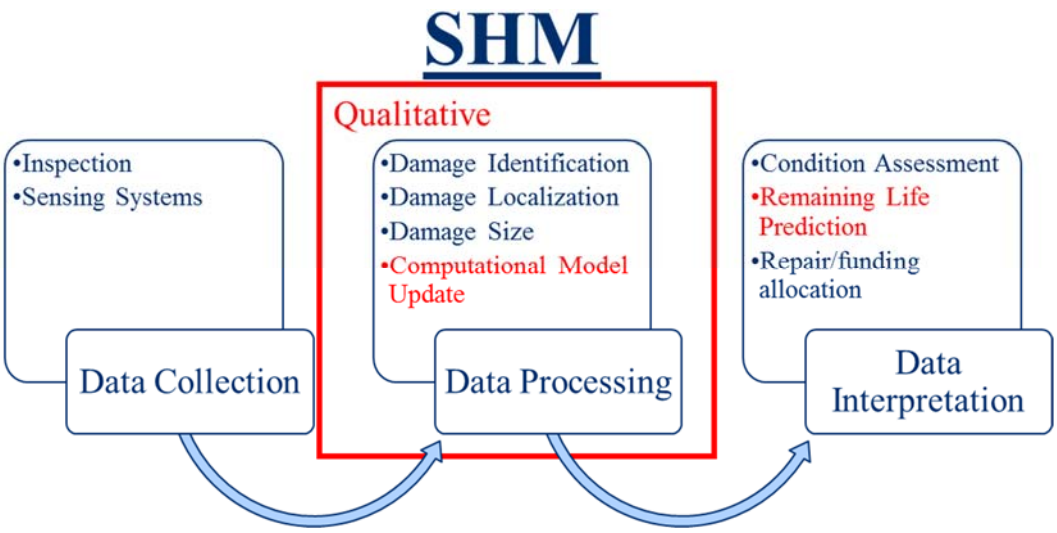


Figure 5. Field practice SHM

Figure 6 shows the technology expected to make an impact on SHM of bridges in the future [5]. The diagram shows how potential new technology can aid in the SHM cycle with better data acquisition, processing, visualization, and analysis. One such platform could be the use of UAS to capture data intelligently. The blue blocks in the figure show

potential segments of the SHM cycle where UAS could be useful as a data acquisition platform and how data captured from UAS imagery can be processed to aid in the decision making and pinpoint areas of interest for future inspections.

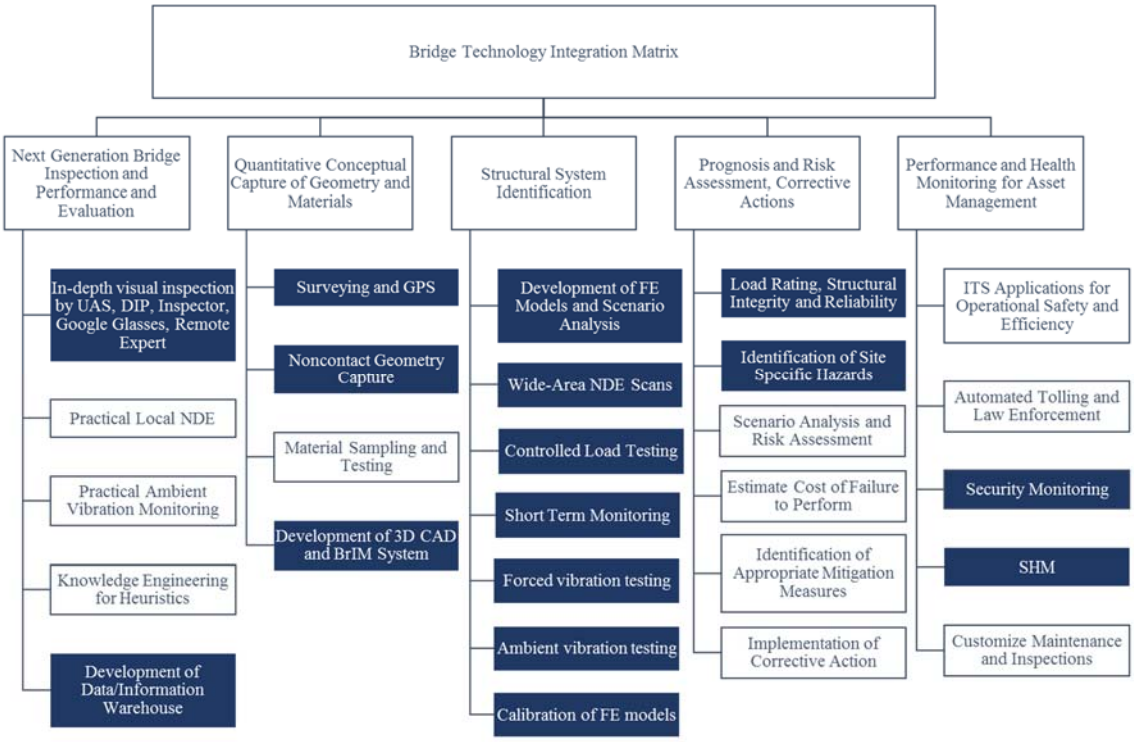


Figure 6. Bridge technology integration matrix for the future of SHM of bridges [35]

Data collection with UAS has already been demonstrated by many researchers proving the systems to be useful as remote sensing platforms. Quantitative geometry has been collected using both LiDAR [36] and structure from motion (SFM) [37]. Damage identification algorithms have been developed for many applications using computer vision, however few of them have been applied to the field due to many limitations including the controlled conditions required, high number of incorrect results, and skepticism of some field experts. It is for this reason that most UAS on the market do not

focus on the data aspect of the system, but on the system's ability to access difficult to reach locations. There is also a large gap in the model construction and interrogation from geometry obtained from UAS. General point cloud data does not include critical information about the material or any information regarding boundary conditions or loading. It is for this reason that bridge information models are very labor intensive to develop [38]. In addition, measurements from UAS imagery is an open area of research [39]. Many researchers have focused on computer vision for a variety of applications and others on photogrammetry, but research on image based measurement using UAS is limited [40].

Computational model updating using experimental measurements has been leveraged in analysis for many years [41-43]. One of the biggest challenges with this technique for in service structures is the accuracy of the model [39]. This is due to the environment in the field including design changes made while building the structure, inaccurate modeling of the ground supporting the structure, inaccuracies in the modeling of friction in the joints, and many others. Furthermore, many in service structures do not have a model, so the model must be constructed.

Images have been used to help create these models, but they lack the ability to understand what the points in the model mean [29, 39]. This complicates matters further because from the point cloud, a user must assign materials, determine the elements in the structure, and sometimes edit the geometry created from LiDAR or SFM [38]. An example of one of the problems that is associated with remotely acquired geometry data is shown in Figure 7 where a bearing is protected by rubber mats to keep out wildlife.

Figure 7. Bridge bearing protection

The rubber mats would completely cover the bearing making it difficult to automatically identify it as a bearing. This is just one simple example of how field conditions complicate the automatic generation of the model. Even after the model is constructed, registration of measured data with the model is another challenging task if the measurement points are not at known locations, such as data acquired from strain or displacement gauges with known locations. Furthermore, if a moving platform is leveraged to acquire data, aligning the two point clouds is more difficult. The iterative closest point algorithm is one example of a way to align point clouds. It minimizes equation (2.1) assuming a rigid body transformation and an initial estimate of the transformation [44, 45].

$$E(R,t) = \frac{1}{N_p}\sum_{i=1}^{N_p}\left\|x_i - R\cdot p_i - t\right\|^2 \tag{2.1}$$

If the initial estimate is poor, the point clouds may not be aligned properly after running the algorithm. The assumption of rigid body motion is a problem for model updating if the experimental measurements show that the object does not move like a rigid body such as bending or fracture. Another challenge with the method is that it requires a

good estimate of the initial transformation which can be difficult to obtain, especially when the point clouds have different densities.

In cases where a model exists, measurements of the physical structure will almost never match the theoretical results of the model and the model must therefore be updated to match the measurements of the structure [46]. Therefore, the model must change through time using experimental data to account for damaged elements which may or may not be identified through inspection, but can potentially be identified by the model through changes in different parameters [47]. If the model is updated and accurate enough to the structure, the model could be used to determine the structural response for simulated loading to determine when damage could be expected to occur [48]. In addition, monitoring a structure through time and updating a model using experimental measurements is a potential way to determine the state of the bridge and leverage this information to provide useful information to owners such as bridge load rating [49, 50]. Both static and dynamic methods for updating the parameters of a model have been applied to different types of structures [51, 52]. The use of finite element models updated with experimental measurements has already started to aid in the SHM cycle for infrastructure and will likely have a larger impact in the near future.

2.2 Unmanned Aerial System (UAS)

2.2.1 Types of UAS

There are four general categories of UAS that most systems fall into: fixed wing, lighter than air, helicopter, and multirotor. Figure 8 shows an example of each of the general type of UAS.

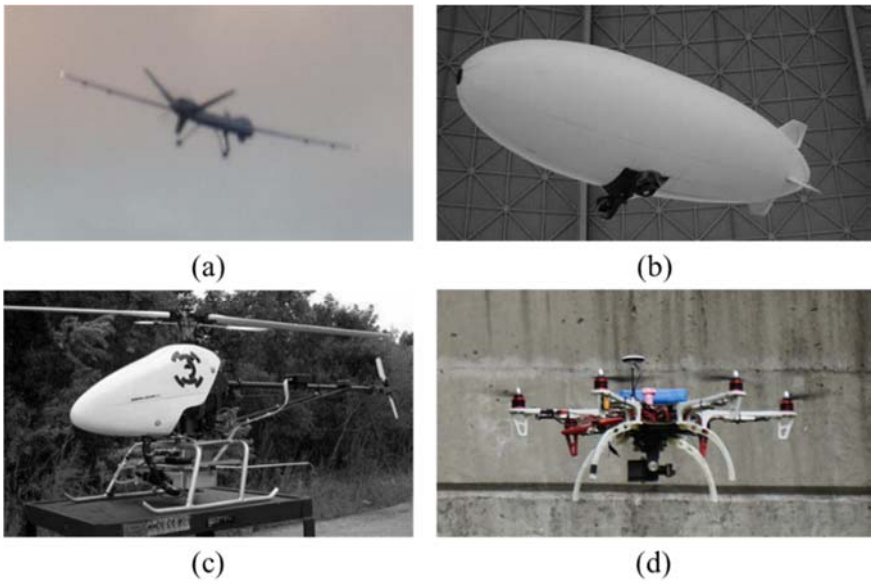


Figure 8. (a) Fixed wing UAS, (b) lighter than air UAS [53], (c) helicopter UAS [53], and (d) multirotor UAS.

Each type has advantages and disadvantages caused by the physical characteristics and the controllability of the system. Lighter than air, helicopters and multirotor systems have the ability to stay in the same place where fixed wing require movement to stay in the air. Fixed wing UAS have the ability to cover large areas very quickly and have high endurance. Lighter than air vehicles have very high endurance, but are difficult to control due to their size, especially in the presence of wind. Helicopter UAS have controllability issues as well which can make them challenging to fly, but they also have much higher endurance and payload capabilities than multirotor systems. Multirotor systems have low flight time and low payload capabilities, but are easy to fly, have redundant motors if there are more than four, and can hover with high stability. UAS applications will have specific challenges associated with them and therefore each type of system should be assessed to determine the best system for the application.

The mechanism for attaining flight as well as the states that are being controlled are also very different for fixed wing, lighter than air vehicles, and vertical takeoff and landing (VTOL) aircraft. The states that are associated with an aircraft are position, velocity, and acceleration in the x, y, and z directions, and angles pitch (θ), roll (ϕ), and yaw (ψ), angular velocity, and angular acceleration of each angle. Figure 9 shows a diagram of the states associated with an aircraft.

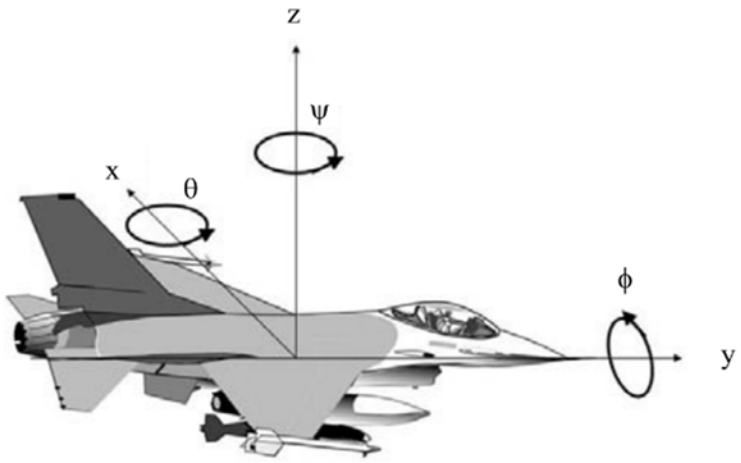


Figure 9. Diagram of aircraft with axes and angles [54]

Only some of these states can be controlled using the inputs of the system. VTOL aircraft require a thrust to weight ratio greater than 1 to fly. Traditional helicopters rely on the tail rotor and the angle of the main rotor to control position and pitch, roll, and yaw of the aircraft. Multirotor aircraft rely on the speeding up and slowing down of different rotors to control the position and pitch, roll, and yaw of the aircraft. The thrust to weight ratio for both lighter than air and fixed wing aircraft can be less than one due to the physics associated with these types of flight. There are several control mechanisms for lighter than air vehicles, but one method could be using rotors to control the position of the aircraft. In

the case of fixed wing aircraft, flaps generally control the pitch, roll, and yaw of the aircraft and the motor controls the thrust.

Figure 10 shows the general schematic of the communication flight control of a UAS with arrows pointing to the direction of the signals. The remote control uses an ultra-high frequency (UHF) allowing the pilot to control the system manually [55]. Different bands of frequency are assigned to specific applications which is defined by the government. Therefore, different countries have different regulations on the frequency allowed for operation of these systems. Most remote controls for UAS use a 2.4GHz frequency with different channels. The more channels the remote and the flight controller have, the more functions the system can perform. Each manufacturer has their own modulation technique to ensure that there is no cross talk between channels. The pilot ground station is not always required for flight, but it can also send a signal using an UHF to the air end telemetry which for instance, controls the system using GPS waypoints. The waypoints tell the UAS to fly through predetermined GPS coordinates at a specified altitude. The ground station can also provide the pilot or observer with information about the system during flight such as remaining percentage of battery, altitude, GPS location, or some types of sensor or system failure [56, 57]. The flight controller constantly receives remote control and/or telemetry and other sensors such as, GPS, compass, and internal measurement unit (IMU). This feedback keeps the system stable. The IMU is composed on accelerometers and gyroscopes. The controller can also receive inputs from other sensors such as LiDAR, ultrasonic sensors, and/or optical flow data for control purposes [58, 59]. The feedback from these sensors is used to send a control signal to the physical components of the UAS, such as servos or Electronic Speed Controllers (ESCs).

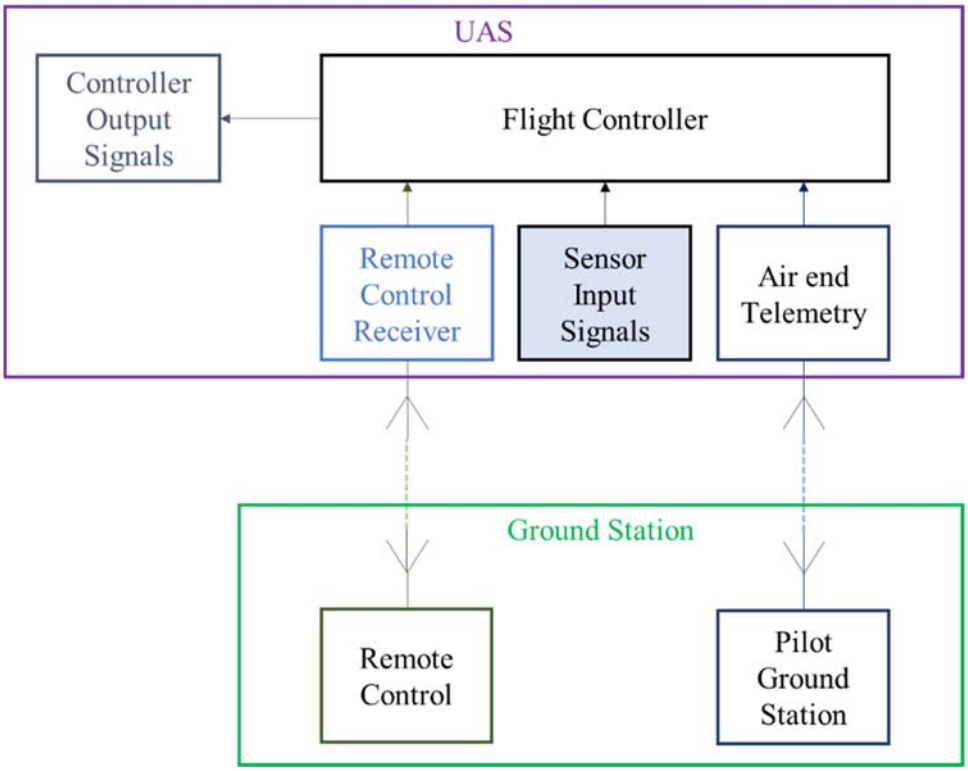


Figure 10. Flight control of UAS

The power system is shown in Figure 11 with the arrows showing the components the batteries power. The battery powers both the power distribution system and the flight controller. In general flight controllers require a step down in voltage because the inputs to the controllers are generally much lower than the voltage required for the motors. The power distribution system ensures that the voltage input to the different components is the same by attaching the different components in parallel. The control signal from the flight controller is used to send the proper amount of current to the physical system to complete the desired maneuver. The feedback from different systems can be used to track the desired output of the system to ensure the maneuver is completed without making the system unstable.

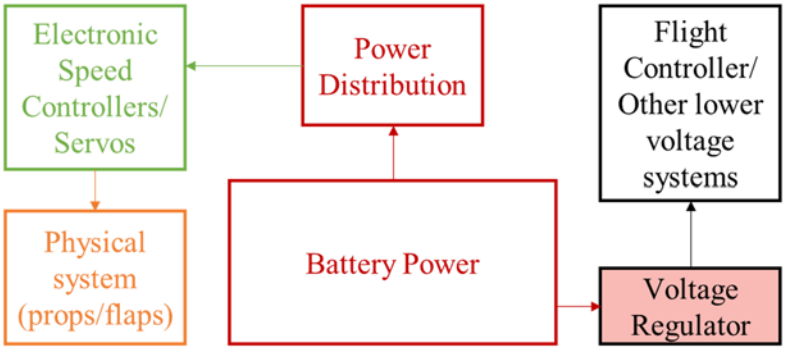


Figure 11. Power distribution of the UAS

Lithium polymer (LiPo) batteries are popular choices for UAS applications due to their fast discharge rates and low weight compared to other battery types. The discharge rate varies with the battery's C rating. Higher values of C result in higher discharge rates which is desirable for UAS applications. The C rating determines the number of amperes (amps) the battery can discharge at any given time. The amps of discharge depend on the C rating and on the size of the capacity of the battery. In the design of the system, it is important to know how much power the system will draw to ensure this value is not exceeded. If the current value is exceeded for an extended period of time, the battery will swell and not hold a charge or the battery could burn up. It is unsafe to use a LiPo battery that has swelled significantly. It is also very important to never discharge the LiPo batteries completely because they will not recharge if they are completely discharged.

2.2.2 Historical Background

UAS have been extensively researched and applied for military applications while several civil applications are currently being developed for remote sensing [60-62]. One of the first attempts of the use of UAS for military applications occurred in 1849 when Austria dropped bombs on Venice using unmanned balloons [63]. This could be questioned as to

the controllability of the aircraft, however this was one of the first successful attempts at using unmanned aircraft for military purposes. Since then, UAS have been used after WWI through today for military purposes expanding from reconnaissance [62, 64]. Today, these UAS are piloted remotely and used for reconnaissance, ground troop support, and eliminating threats. According to David Deptula, "we are with UAS in the same place we were with biplanes in WWI [65]."

The acquisition of data has always been a goal of aerial spy aircraft. With this in mind, the UAS primary objective is to collect data more efficiently knowing that the data is the only output that matters. The acquisition of data is therefore one of the most important objectives for non-military use. Since being developed and implemented for military applications, several non-military uses have emerged from hobbyists, to delivering packages, aerial imagery and film, to inspection and monitoring applications. The advancement of this field has produced a new draw towards multirotor UAS for many different applications due to their ease of operation, low cost, and ability to hover in place. There have been concerns about the use of UAS in the national airspace and the Federal Aviation Administration (FAA) has started implementing and enforcing regulations to ensure the reliability and safety for all aircraft.

2.2.3 Regulations for Hobby and Commercial Use

Congress has tasked the FAA with opening up the United States National airspace to commercial UAS [66, 67]. Since then, the first Certificate of Authorization (COA) was granted to the oil company BP which allowed the Puma AE UAS to conduct oil pipeline inspections in Alaska [68]. Furthermore, the FAA had selected six "Center of Excellence" research test sites for UAS [69]. These events have created many new possibilities for

civilian applications of UAS including infrastructure visual inspection and regulations on the use of UAS have been developed to allow the industry to expand while keeping American airspace safe. Figure 12 shows a list of some of the significant events starting in 2012. Most of the state legislation that was proposed to regulate UAS was based on privacy and safe operations around specified areas as well as punishments for breaking the laws [70].

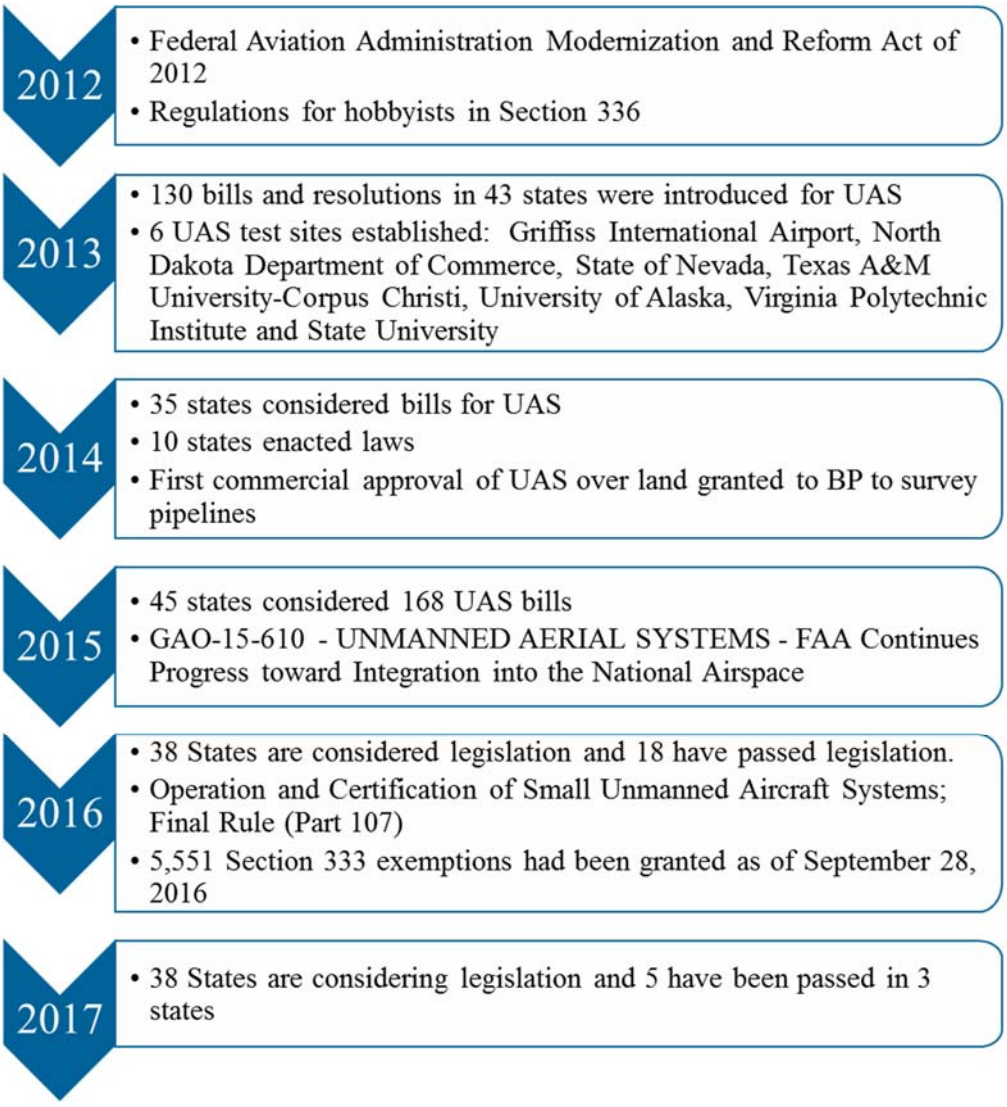


Figure 12. Timeline of regulations and steps to UAS integration into United States National airspace [67-74]

The use of UAS in American airspace is restricted to hobbyists who are registered with the FAA for any aircraft over 0.55 pounds (~250g) and under 55 pounds (~25kg) [73]. This law went into effect in December of 2015 and as of May 12, 2016, over 450,000 UAS were registered by hobbyists in the United States. A map showing the number of UAS registered per state is shown in Figure 13 [75]. The rules for hobbyist are the user must be at least 13 years old to register, fly below 400 feet, fly within visual line of sight (VLOS), and do not fly near other aircraft or within 5 miles of an airport without contacting the airport and agreeing on rules to follow. Furthermore, the use of UAS under the influence of drugs or alcohol, flying at sporting events, over people, near other aircraft, or near emergency situations is strictly prohibited. The pilot's Small UAS Certificate of Registration Number must be permanently put the UAS. Hobbyists do not have to follow all of the requirements under Section 107, but do need to follow the laws set forth in section 336 [67, 73].

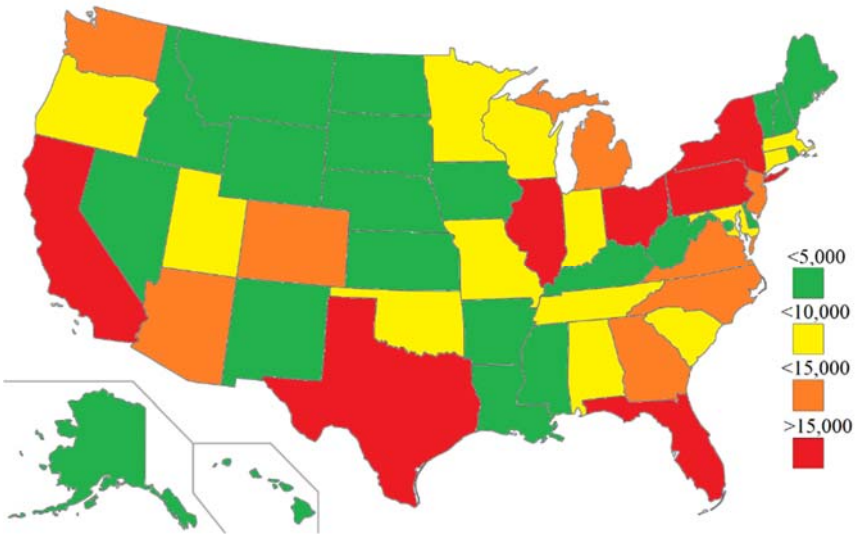


Figure 13. Number of hobbyist UAS registrations by state as of May 12, 2016 [75]

Up until the rules for small UAS was released, civil organizations were required to complete a Section 333 Exemption to fly UAS with more strict rules than hobbyists. This was a step to allow companies to use UAS before the rules were released. Section 333 allowed the companies to fly UAS by exempting them from rules used for general aircraft in the national airspace once the petition was approved. There were over 5,500 Section 333 exemptions granted by the FAA by September 9, 2016 [71]. The rules for small UAS were a more permanent solution and were released on June 28, 2016 [74]. Under Section 107, all pilots must pass the remote pilot knowledge test and vetted by the Transportation Safety Administration (TSA) and obtain a remote pilot certificate [76]. The reasons for the intermediate steps was to satisfy the goal of integrating UAS into national airspace safely while still allowing the expansion of the UAS market as quickly and efficiently as possible.

In cases where the rules for small UAS cannot be followed, for instance, if the UAS take-off weight was greater than 55lbs, a section 333 exemption would still be required. A certificate of authorization would also be required after the exemption was granted and an airworthiness certification is also required [71].

2.2.4 Commercial Multirotor UAS and Software

Many companies have emerged in the hobbyist and commercial UAS markets. DJI is the largest supplier of UAS for commercial and hobbyist applications. Other companies including Lockheed Martin, SenseFly, 3D Robotics, Parrot, Asctec, CyPhy, Aeryon Labs Inc., Microdrones, and many more have developed for commercial and industrial applications of UAS for data collection. Most UAS come with ground control station software that can be used to program waypoints, determine the state of the UAS for battery life, speed, altitude, etc., and program different outputs for a variety of sensors and other

onboard systems. Some of the ground control software is open source such as Qground Control and MissionPlanner. They are often used as research platforms with a variety of flight control hardware [57, 77, 78]. An example of ground station control used during an automated flight is shown in Figure 14. The flight data is boxed in orange, the waypoints are highlighted with the teal arrow, the red arrow shows the home point, the green arrow was the current UAS location, and the yellow arrow was the distance between the two waypoints. The final landing position is show with the black arrow.

Figure 14. DJI ground station software

In general, all of the flight control software has the elements shown in Figure 14 to aid in the mission control. More sophisticated systems exist to aid in the flights that provide better accuracy measurements and more control of the system. The use of these systems aids in flight control and allows the pilots to focus more on ensuring safety rather than actually flying the aircraft. In the case where data is acquired, the pilots can focus on data acquisition while monitoring the flight.

There are four general forms of flight control used for UAS: manual, stabilize, position hold, and automated. Manual mode uses no sensors and therefore, this mode is the least used and last resort flight control in almost all cases because multirotor aircraft are unstable systems that rely on the flight controller and input from the sensors to keep the system stable. Manual flight mode does not use many of the sensors and is generally reserved for when systems are failing or in the case where an experienced pilot intends to perform complex maneuvers. Stabilize flight mode uses all of the stabilization sensors and sometimes an altitude sensors to keep the system level during flight. The system does not have any position feedback to keep the system in the same location. Position hold uses the same sensors as in stabilize mode in addition to a position sensor used to keep the UAS in the same location. This position sensor can be GPS, optical flow, motion capture, or any other sensor that measures the position or relative position of the system which is used as an input for a flight controller a tracking algorithm. Finally, automated flight mode uses the sensors on the system to conduct a predetermined flight path. In general, GPS is used with preprogrammed waypoints to complete the flight mode, but methods involving 3D simultaneous localization and mapping (3D SLAM) to map and navigate through the environment have been demonstrated [79].

Most of the UAS data processing software is limited to mapping and modeling of geometry. Precision Hawk, DroneDeploy, Skycatch, Pix4D, Pix Processing, 123D Catch, and ContextCapture, are just a few examples of software that use images to create 3D models. This type of software is critical to the understanding of the data acquired from the camera systems. Without the global context, the images are not nearly as useful. There are companies, such as Nexco, that perform semi-automated damage identification on bridge

decks using ground vehicles [80]. However, automated damage identification from images is an open area of research. Commercial software for identification of damage on structures is the next step to UAS providing useful information to decision makers.

2.2.5 Civil Applications

The use of UAS for civil applications is a much more recent topic of research and many different uses have been developed [81]. UAS have been used for mapping [82], monitoring of difficult to access areas such as volcanos [83], search and rescue [84], crop monitoring [85], and inspection [11, 61]. UAS inspection is an open topic of research for different types of infrastructure systems including buildings [86, 87], pipelines [88], powerlines [89], and bridges [90]. The UAS sensing systems are generally limited to noncontact systems including LiDAR [91], color imagery [92], IR imagery [93], and wireless sensors [94]. Recent research has also investigated the use of UAS for contact interrogation with impact hammers [95].

Some of the major advantages of UAS platforms being leveraged to perform inspections is their ability to access difficult to reach areas of the structure, carry multiple payload types, the collections of GPS and IMU data, and the high resolution full field data the sensing systems acquire. The most common inspection accidents involve the use of ladders which are used to gain access to different areas of the structure [96]. UAS also have the potential to decrease the risk to inspection personnel. Some major concerns involving UAS inspection are the low battery life, unreliable positioning and lack of collision avoidance in complex field conditions, environmental conditions on a structure, limited payload, and missing data. One of the biggest challenges is the environmental conditions around civil structures because they cause difficulties in UAS navigation and different

methods have been proposed to mitigate these issues, but there are no effective methods currently on the market today that can fully cope with the challenging conditions around civil structures [97, 98]. Like all forms of nondestructive testing, UAS cannot be the only tool used to solve all of the inspection related problems. However, they do have the potential to serve as the first pass of rapid inspection to target areas of interest for other types of nondestructive testing techniques. Contact sensors applied by inspection personnel or on other robotic platforms, such as the Robotic Assisted Bridge Inspection Tool (RABIT^{TM}), have advantages over noncontact sensors in terms of accuracy, however these techniques are slow and require traffic disruption [20]. The UAS could be employed as a first pass of inspection to identify areas of interest on the structure for inspection personnel or a system such as RABIT^{TM} to complete a more in depth, accurate assessment of the structure without disrupting traffic for long durations of time.

The sensing systems that are most widely implemented in conjunction with aerial systems are cameras. Cameras are lightweight, data rich, and comparable to what an observer would see during the flight. Color imagery can be used to identify surface problems on infrastructure systems, however, the analysis of these images is an open area of research. For any application involving remote sensing that result in large datasets, it is crucial to eliminate redundant and insignificant data without removing important information. Many image processing algorithms have been used to identify objects from images to achieve this objective. For instance, images have been used to identify the spalling of concrete bridge elements [99, 100], a robot-mounted machine vision system has been used to identify cracks on a bridge [101], and images obtained at night with different light sources have been used to identify roadway surface cracks and other elements such

as manholes and spills [102]. Further, digital image correlation (DIC) approaches have been utilized to analyze in service structures under loading to estimate displacement and vibration responses [103, 104]. Aerial inspection using fixed-wing aircraft has been used for feature identification of linear infrastructure such as roadways to identify potholes and cracks [105, 106]. Color has been used in conjunction with machine learning to classify images with rusted areas on bridges using a robotic platform [107]. Hough transform was used to identify lines corresponding to the rows in vineyards from UAS images to find unhealthy plants [108]. Edge detection was used for crack identification in building facades using UAS imagery [109]. Moreover, a region growing technique was leveraged to identify delaminations from IR images of a manufactured concrete block with known defects [110]. Furthermore, localization is another important aspect of navigating through large sets of data. To help solve this problem, a 3D building model was constructed using a multirotor UAS by stitching images together and projecting them onto known geometry [111]. Structure from motion (SFM) was used to construct a 3D model of a retaining wall leveraging many images acquired with a UAS to provide a high accuracy dense reconstruction [86]. Similarly, a 3D point cloud reconstruction of a pedestrian bridge was completed by Lattanzi et al [37] using a dense SFM method applied to UAS imagery [28]. Despite the challenges associated with implementing SFM from UAS imagery, this method could provide a global view of the structure as a tool to identify and locate damage as well as provide useful context to the owners of the structures.

Another popular sensor that can be utilized in conjunction with UAS are LiDAR and other 3D scanning technologies. A laser scanner and RGB-D sensor were used as the payload on a UAS to obtain point cloud information of outdoor buildings for inspections

purposes [91]. Furthermore, 3D SLAM was implemented to model buildings and streets using mobile vans using laser scanners, global positioning systems (GPS), cameras, position orientation systems (POS), and internal measurement units (IMU) [112, 113]. 3D SLAM is used for localization and navigation [114], environment mapping [115] and 3D surface geometry mapping [116]. 3D SLAM applications using a UAS platform for outdoor environments has been demonstrated [114, 117], which is promising for applications of the 3D SLAM approach on infrastructure systems. Terrestrial laser scanning was used to collect a 3D point cloud of a bridge structure that was used for localization and visualization of the structure on a computer [36]. The 3D measurement systems could potentially obtain global measurements of a structure as well as help the UAS with navigation making it a promising technique that could further aid in the monitoring of civil structures as their accuracies improve.

2.3 Image Processing Algorithms

2.3.1 Filtering

Filtering is very important to eliminate parts of the image that are unwanted and highlight parts of the image that are desired. There are many different types of filters with different properties that achieve different results. For instance, linear filters are fast, achieve many objectives, and are easy to implement by convolving a window with the image. Convolution is defined for discrete images using equation (2.2) [118]. The convolution window (w(s, t)) is moved across the entire image (f(x, y)) and the elements of the window are multiplied and added together to create the filtered image g(x, y) for all pixels in the image. This is effectively applying a transfer function to the image.

$$g(\mathrm{x},\mathrm{y}) = \sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)\, f(x+s, y+t) \tag{2.2}$$

Some examples of this type of filter are low pass filters such as averaging filters or Gaussian filters, edge detection such as Prewitt or Sobel edge detection, and sharpening or edge enhancement such as difference of Gaussians or Laplacian of Gaussians. Low pass filters have the property of the sum across the filter is equal to one. Edge detection or high pass filters generally have the property where the sum across the filter is zero. Edge detection methods can also have elements that aid in detection such as the Canny edge detection method which uses the strongest edges to help identify connectivity to the weaker edges [119]. One of the problems with linear filtering comes at the edges of the image. Assuming the filter has an odd dimension (3x3 for example), the filtering at the top left corner pixel of the image does not have enough pixels to complete the filter. This problem is shown in Figure 15 where the blue pixel is the pixel being analyzed, the green pixels are the pixels in the filter that are in the image, and the red pixels are outside the image border. One way to handle this problem is to neglect the edge pixels completely, but this results in a smaller image. A boarder of zeros can be added to the image, but this will affect the boarders of the image in the resulting image. The other option is to repeat the edge pixels around the boarder and the last option is to mirror the border of the image to account for the missing pixels. The last two options achieve better results, but the edge pixels still cannot be relied on completely because they are based on simulated values.

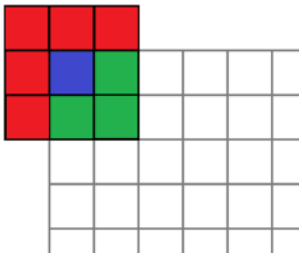


Figure 15. Linear filter of an image at top left corner pixel.

Gradient are also a useful tool for image analysis. The gradient of an image is simply the difference of two pixels across the entire image, similar to convolution. There are several methods to complete this, but generally, an edge detection window is use and instead of setting a threshold to detect the edge, the resulting value is stored in a new image. The gradients can be used for many different applications such as steering a segmentation algorithm or adaptively thresholding an image using the magnitude and angle of the gradient [118].

Nonlinear filters can achieve different results than the standard linear filters and they cannot use equation (2.2). The median filter is appropriate for removing salt and pepper noise. It takes the median value associated with a local area in the image and this value is placed in a new image. Similarly, maximum and minimum filters are used in the same way only instead of saving the median, they choose the maximum or minimum. Depending on the type of noise, low pass filters or median filters can produce better results, but both accomplish the same goal of denoising the image. Adaptive filters are also capable of removing noise from the image. In general, they are more computationally expensive, but they can provide very good results. They use a local region of pixels to determine how to filter the image using gradients, intensity values, and/or other criteria such as the location of the boundaries to determine what type of filtering to use. This type of filter will generally have to be created by the user for a specific application, but the results are often worth the computation cost and the time for development [118].

2.3.2 Morphological Operation

Morphological operations are usually performed on binary images. The methods use a structuring element to determine the degree in which they change the image. In

grayscale images, dilation and erosion work as a max and min filter respectively and he largest or smallest value within the structuring element is kept for the new image. In the case of binary images, there are several more operations that can be completed including dilation and erosion as well as hole filling, skeleton, and boundary extraction. The hole filling algorithm is used to fill the holes that appear in blobs of a predetermined size. An image skeleton is erosion of the image until only one pixel width is left in each blob. There are several other types of morphological operations that are used on binary images to make identification of objects easier [118]. An example of image dilation is show in the top row of Figure 16 and erosion is shown in the bottom row.

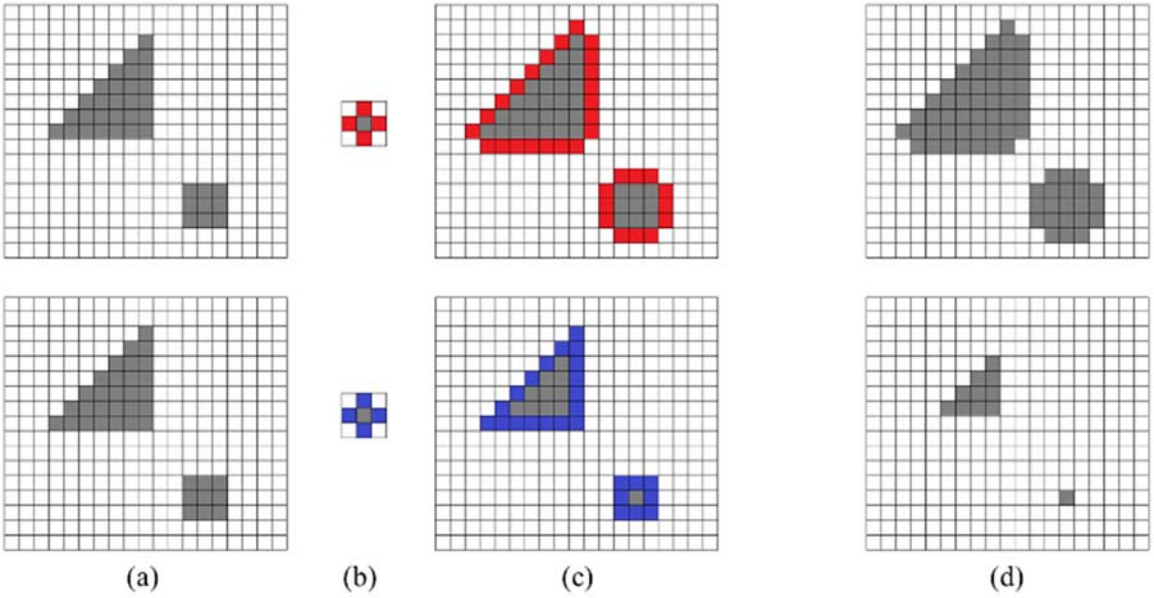


Figure 16. Dilation of the image (top row) and erosion of an image (bottom row). (a) Original image. (b) Structuring element. (c) Pixels effected. (d) Resulting image.

The original image is shown in (a). The structuring element in both images is the same, but the pixels being operated on are different as shown in (c). The red pixels are

added and the blue pixels are removed. Figure 16 (d) shows the resulting image after dilation and erosion.

2.3.3 Histograms

Image histograms are useful for many applications, but one of the most important is thresholding. Many different methods can be applied to threshold images, but often histograms can be used to find natural points appear to be good threshold values. One method to help identify these points is to fit several Gaussian distributions to the data. A good way to think about an image histogram is a probability mass function (pmf) for the image. If all of the histogram are added together, this number can be used to normalize the histogram to make it satisfy the rules of the cumulative mass function (cmf) starting at 0 and ending at 1. Furthermore, histograms can be used to normalize the intensity throughout the image. This can be accomplished by selecting the intensity value and moving up to the corresponding value on the cmf and setting the intensity values associated with that bin to the value of the cmf. This will make all of the values in the new histogram have roughly equal counts in each bin. Similarly, contrast limited adaptive histogram equalization uses the same idea, but breaks the image into blocks and computes the histogram equalization. Then, it compares the values to the neighboring blocks to ensure there are no dramatic changes in contrast along the boundary of the block [118]. An example of these techniques is shown in Figure 17. The images are on the top row and their corresponding histograms are shown below. The original image is shown in Figure 17 (a). After histogram equalization, the histogram in Figure 17 (b) shows all of the values have equal counts if you distribute them over the entire range of values that color encompasses. Finally, in Figure 17 (c), the adaptive histogram equalization shows the histogram is not equalized in

the same way, but more features are clearly visible in the image where they were less visible in the original and the histogram equalized image.

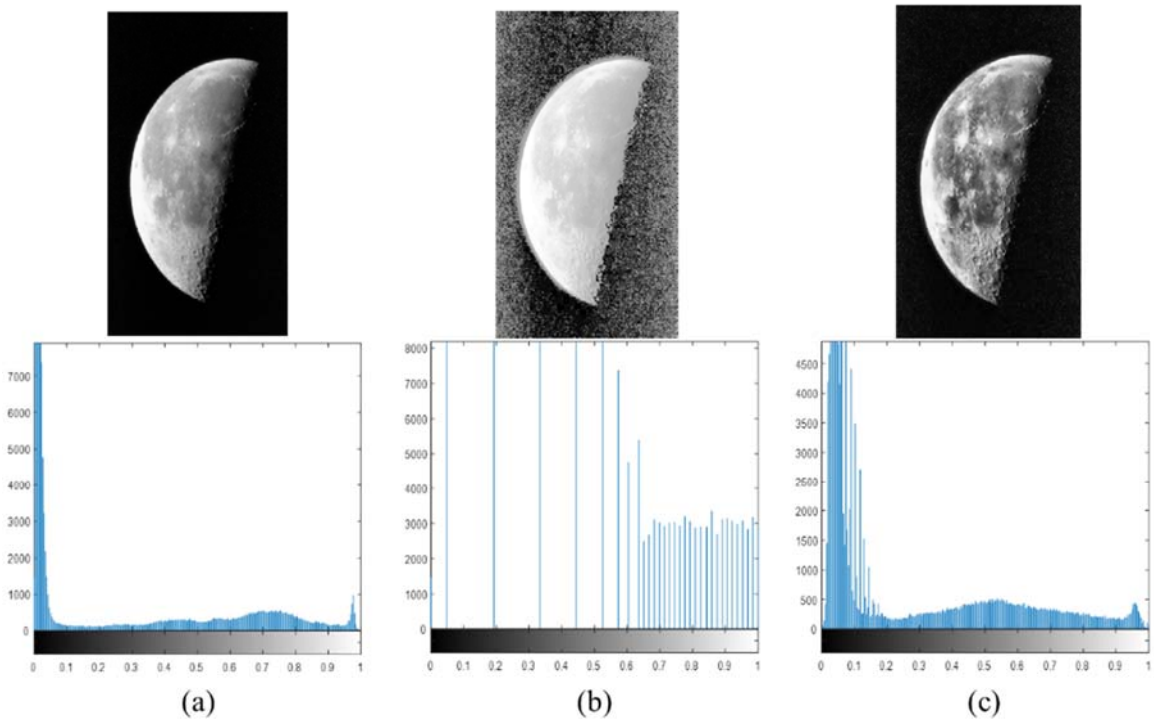


Figure 17. (a) Original image with corresponding original histogram [120]. (b) Histogram equalized image with corresponding histogram. (c) Contrast limited adaptive histogram equalization and corresponding histogram.

2.3.4 Segmentation

One of the simplest ways to segment an image into foreground and background is thresholding [118]. This technique is rarely sufficient to truly segment the image because segmentation is used to understand what is in the image, therefore thresholding is often combined with other more sophisticated techniques. It is often difficult to determine what the threshold values and justifying the choice of that threshold. One way of choosing a threshold is the Otsu method, which is an automatic way for assigning an optimal threshold [121, 122]. This method is not limited to image analysis, but does work directly on the

histogram which makes this a very fast method for optimal thresholding. It minimizes the variances within classes and maximizes the variances between classes to form the optimal threshold. This method is also not limited to two classes and has been extended to multiple classes. As described in section 2.3.3, Gaussian distributions can also be used to describe an image histogram. Using the crossings of the Gaussian distributions, an optimal threshold can be set to distinguish what parts of the image are foreground and background. In both cases, the user does not have to manually select a threshold.

Texture in an image can also be used to segment an image. This method can use entropy, variance, higher order moment methods, and several others [99, 123]. These methods focus on blocks within the image to compare a metric or metrics associated with the texture of the image. Morphological operations can be used to segment by texture as well. If an image is expected to have blobs of a specified size and spacing, proper choices of structuring elements used for dilation and erosion can help segment the image [118]. The changes in intensity within the blocks can aid in the decision making for what type of texture is associated with different areas of the image. Orientation of intensity changes are also used to classify texture [124]. There are several databases that are easily found online of different images and different types of textures that can be used for automated identification [125].

The use of intensity values and gradients can also be used to segment an image by following the path of least resistance in the image. Gradients can be calculated in many ways for images and filters similar to the edge detection filters presented in section 2.3.1. Methods for obtaining the gradient can also be determined using subtraction with the central difference method or the intermediate difference method. The gradients are

calculated in both the horizontal and vertical directions which are used to calculate the magnitude and the direction of the gradient. These parameters can be used to follow an outline of an object to segment it from the image [126].

Region growing is another method that can be used for segmentation. It is related to the morphological operations, but it is not bounded by a structuring element and has a criteria that causes it to stop growing. It requires seed points and a stop criteria which could be a probability models based on intensity value, gradients, or change in textural features [127, 128]. This method has been very useful for many image segmentation applications from medical imaging to civil infrastructure to identify cracks and delmainations [110, 126, 129, 130].

Active contours or morphological snakes is a method that uses gradients, color values, and the stiffness of the spline to minimize an energy equation [131, 132]. The major drawback to this method is the need for an initial guess at the contour boundaries with ordered points, however once this is achieved, the algorithm converges to boundaries very well when the correct parameters are selected. The energy minimization takes the form shown in equation (2.3).

$$E = \int_0^1 E_{Int}\left(v(s)\right) + E_{Image}\left(v(s)\right) + E_{Con}\left(v(s)\right) ds \tag{2.3}$$

The three terms are the internal energy of the spline ($E_{int}(v(s))$) [132], image energy ($E_{image}(v(s))$), and other energy constraints $E_{con}(v(s))$. The v(s) term is the ordered points for the initial boundary. The internal spline energy corresponds to the stiffness of the spline and is determined using equation (2.4) [132]. This term is based on the norm of the first and second derivatives of the spline function. The image energy is described in equation (2.5) [132] where the line terms push the spline towards lines, edge term pushes the spline

towards edges, and the termination (term) term forces the spline towards corners and bends [132]. The other energy constraints could be used to avoid certain areas of the image or force the spline in a different direction when it gets close to predetermined features, but this term will change with the intended purpose.

$$E_{Int} = \frac{\alpha \left| \frac{dv(s)}{ds} \right|_2 + \beta \left| \frac{d^2 v(s)}{ds^2} \right|_2}{2} \tag{2.4}$$

$$E_{Image}\left(v(s)\right) = w_{line} E_{line}\left(v(s)\right) + w_{edge} E_{edge}\left(v(s)\right) + w_{term} E_{term}\left(v(s)\right) \tag{2.5}$$

The α, β, and w terms are weighting functions that are set by the user. These terms could be functions or constants that heavily affect the performance of the method.

Hough transform is a technique that uses votes to identify a defined shape [133]. The shapes this technique is most commonly used to identify are lines and circles. This is because the search space for these shapes is relatively small. The technique is used to cast votes to a space based on a binary image. The transform uses the equation for a shape and casts votes based on the space. In the case of lines, the space is determined through the distance and the angle to each white pixel to the origin of the grid. Each pixel votes at every step using equation (2.6), where x and y are the coordinates of the white pixel and theta is varied from -90 degrees to 90 degrees.

$$\rho = x\cos(\theta) + y\sin(\theta) \tag{2.6}$$

In order to identify the lines, the locations in the Hough space (r, θ) with the highest number of intersections has the highest number of votes. This can be seen in Figure 18 where the original image and set of points are shown in (a) and (b) respectively. It should be noted that the y-axis is flipped in Figure 18 (b) because they coordinate system then

matches the pixel coordinate system shown in Figure 26. The Hough transform is shown with the intersection of the three curves highlighted in red, roughly at ρ = 0 and θ = -45˚.

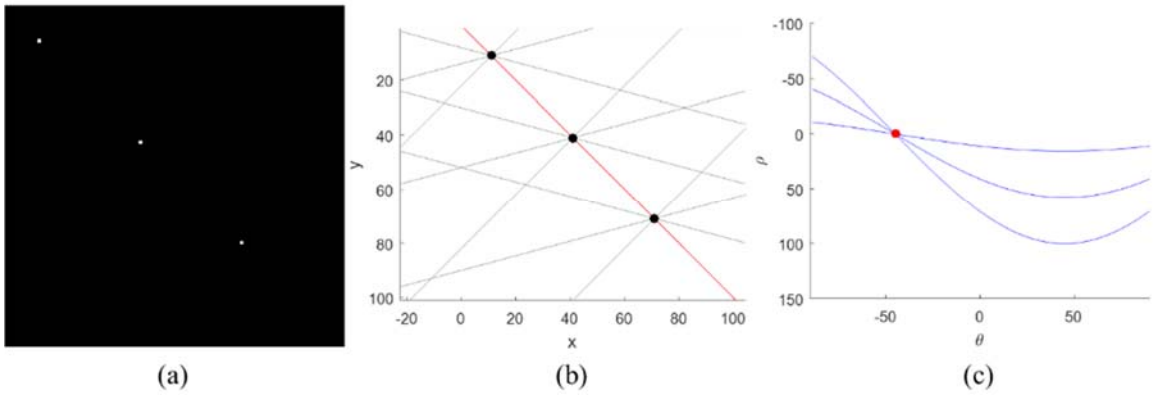


Figure 18. (a) Original image. (b) Original set of points. (c) Hough transform of the three points.

Other shapes can be identified using Hough transform by fitting a different equation, but the parameter search space increases as well. In all cases, the location in the Hough space that receives the highest number of votes has the strongest correspondence to the predefined shape. The major advantage of Hough transform is the entire shape is not required to appear within the image for the shape to be identified and it is also robust to noise [128]. This is important for recognition because it will miss fewer shapes, however it is also important to note that if the shape varies significantly, the technique does not as work well [133].

Another method used for image segmentation to identify bright areas in images is the peaks over threshold method which is very robust to noise compared to other methods of segmentation. This method has been used to identify proteins in fluorescent labeled cells [134]. The image is modeled as low frequency background, noise, and the peaks. A

simulated image was created and displayed in Figure 19 with each part of the image separated and combined.

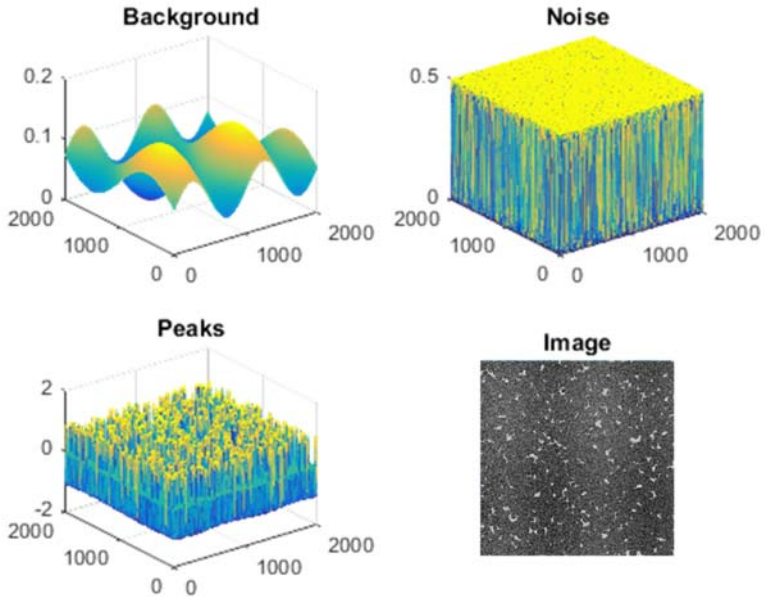


Figure 19. Image model for the peaks over threshold method

The method relies on a conditional probability to determine if the peak is high enough to exceed the threshold given the criterion associated with the distribution defined by the user. First, an initial threshold is set to identify values that could potentially be accepted as peak. The distribution used for this method is shown in equation (2.7) [134].

$$F(x)=\begin{cases} 0 & x<\tau \\ 1-e^{-\left(\frac{x-\tau}{\theta}\right)} & x\geq\tau \end{cases} \tag{2.7}$$

In the equation, τ is the initial threshold, θ is defined in equation (2.8), and x is the value in the image [134].

$$\theta=F^{-1}\left(\frac{1}{2}\right) \tag{2.8}$$

The final threshold used to segment the image is defined in equation (2.9) where $\hat{\tau}$ is a second user defined threshold [134].

$$\gamma = \tau - \theta \cdot \log\left(1 - \hat{\tau}\right) \tag{2.9}$$

A main advantage to using probabilistic methods for determining segmentation threshold values is that it is based on a model to determine what points are large or small enough to be determined significant. For cases where the foreground is expected to be much smaller in number of pixels, for instance, in the case of cracks, the peaks over threshold method would be expected to perform well. Clustering methods are similar to other probabilistic methods, but instead of setting a threshold, they minimize a function to obtain the segmentation [135]. There are several different methods to cluster an image using different properties including texture features, color features, shape features, and many others. K-means is a common method of clustering used on many different data types including images. For this algorithm, k centers are randomly set and the classes are minimized using the error equation (2.10). E is the error, x_i^k are the data points, n is the number of data points, c_k is the center, and K is the number of clusters [136].

$$E = \sum_{k=1}^{K} \sum_{i=1}^{n} \left\| x_i^k - c_k \right\|^2 \tag{2.10}$$

The process is repeated until the location of the centers converges. After convergence, all of the data points are placed into the cluster with the closest center. Fuzzy clustering is essentially the same algorithm, but the points are not assigned to a specific cluster. Instead, they are assigned weights to each cluster to determine the proximity to each cluster center. Both types of clustering can be applied to different applications depending on the desired output.

2.3.5 Keypoint Identification and Matching

Keypoints in an image are points with unique characteristics that can be identified easily using some metric. These points have been determined many different ways, but some of them include corner detectors and local extrema detection [137]. The metrics used to identify these points can be intensity values, gradients, or other pixel based features, as well as how these features compare to their neighbors. One popular method that is often used for keypoint identification due to the robustness is scale invariant feature transform (SIFT). This method finds local minima and maxima using a multiresolution difference of Gaussian filter to create a 128 dimensional feature vector used to match points within images [138]. Each point can then be compared to points within another image to determine if the points are a match. Some other methods for keypoint identification and matching include Speeded Up Robust Features (SURF) [139], Features from Accelerated Segment Test (FAST) [140], Oriented FAST and Rotated BRIEF (ORB) [141], and many others [142]. Matching or corresponding points can be used for several tasks including geometry estimation, sensor movement, and object identification based on a template [138, 143].

2.3.6 Blob Features and Classification

After an image is converted to a binary image, blobs appear as groupings of black or white pixels. These pixels can be described using many different features. Some of the features include perimeter, area, bounding box, major and minor axes of the bounding ellipse, and many others [120]. They can be combined as well to produce features like circularity ratio as calculated by equation (2.11) where A is area, P is perimeter and R is circularity ratio. The circularity ratio has also be called the thinness ratio, roundness, or shape factor [118].

$$R = \frac{4\pi A}{P^2} \tag{2.11}$$

Some other features include: convexity (equation (2.12)), adjusted roundness (equation (2.13)), aspect ratio (equation (2.14)), equivalent diameter (equation (2.15)), and eccentricity ((2.16)) [120].

$$C = \frac{A}{A_{convex}} \tag{2.12}$$

$$R_A = \frac{4\pi A_{convex}}{P^2} \tag{2.13}$$

$$A_R = \frac{A_{major}}{A_{\text{minor}}} \tag{2.14}$$

$$E_D = \frac{4A}{\pi} \tag{2.15}$$

$$E_C = \frac{\sqrt{A_{major}{}^2 - A_{\text{minor}}{}^2}}{A_{major}} \tag{2.16}$$

Descriptions of perimeter (P), area (A), convex area (A_{convex}), major axis (A_{major}), and minor axis (A_{minor}) are shown in Figure 20.

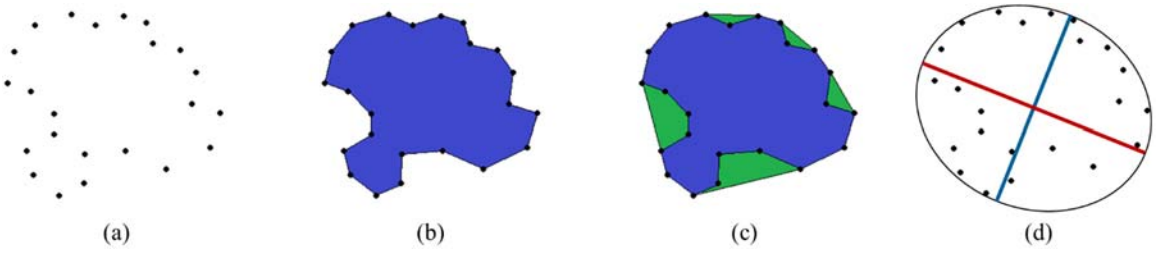


Figure 20. (a) Group of points surrounding a blob. (b) Perimeter (P) in black and area (A) in blue. (c) Convex area (A_{convex}) in green and blue. (d) Major axis (A_{major}) in red and minor axis (A_{minor}) in blue.

In general, features that are dimensionless perform better for analysis because they do not rely on the scale of the image which is why circularity is desirable compared to area or perimeter. Dimensionless features generally involve ratios of features with dimension and include the ratio of area divided by convex area and ratio of minor axis divided by major axis of the bounding ellipse.

Classification can be completed using many different machine learning algorithms for image analysis [136]. There are two types of machine learning algorithms: supervised and unsupervised. In general, supervised learning has a predetermined number of classes and a training and test set are used to determine the algorithm's performance. Unsupervised learning is used to cluster a dataset using the natural differences in the dataset. These methods do not have a predetermined outcome. One example of unsupervised learning is the k-means algorithm described in section 2.3.4. Examples of supervised learning include support vector machines, k-nearest neighbors, and decision trees [144]. There are several different software packages which allow a user to test the performance of different machine learning algorithms to determine which will work best for the application [120, 145]. The major advantage of unsupervised learning is the outcome does not rely on user inputs prior to classification. The disadvantages are that this method is not consistent causing convergence to different centers and the outcome classification is not based on a physical description. The advantages of supervised learning are the outcomes are based on a physical description determined during the learning stage and the methods will classify the data more consistently [146]. The description of the objects in the image are broken down into features which are analyzed by the machine learning algorithm. Identifying intelligent features is one of the most important steps in distinguishing between classes using machine

learning. Another important step is determining what machine learning algorithm will be appropriate for the application. Different algorithms can be used for unsupervised and supervised learning and their performance varies based on the features chosen and how they can be separated using the equations associated with each method. In general, machine learning algorithms try to minimize some property within the data. The k-nearest neighbor approach uses the location of the new data in the n-dimensional space and bases the classification on the closest neighboring point. The k-means algorithm tries to minimize the Euclidean distance of all of the features from the cluster centers. Decision trees, support vector machines, artificial neural networks, and several other methods have been developed for machine learning and each one has its advantages and disadvantages [147]. When making the decision on which algorithm to choose, the speed of training and learning, accuracy, and robustness to different problems such as noise or overfitting should be considered as well as the type of separable data such as linear or nonlinear [136, 147]. There are several metrics that can be used to determine how well the classification performed. True positive and true negative are the conditions where the ground truth and the prediction are the same class. False positive and false negative are the conditions where the ground truth and the prediction are not the same. These values can be used to determine the false positive rate, false negative rate, true positive rate, and true negative rate as well as construct a receiver operating characteristic curve or determine other metrics such as precision and recall [144].

Machine learning and many segmentation methods rely on how easy it is to separate the data in question. Threshold based segmentation that relies on an optimal threshold or Gaussian mixture models used to identify a threshold need adequate separation of the data.

In general, the preprocessing and filtering techniques are used to make the data more separable. A simple example of the ability to segment data into two classes is shown in Figure 21. Figure 21 (a) and (b) show two distributions of data with different separation between the means and the same standard deviations. The optimal threshold is shown as a black line. Figure 21 (c) and (d) show the correct and incorrect classifications. It is clear that the larger separation between the means causes a much better separation with fewer mistakes [148].

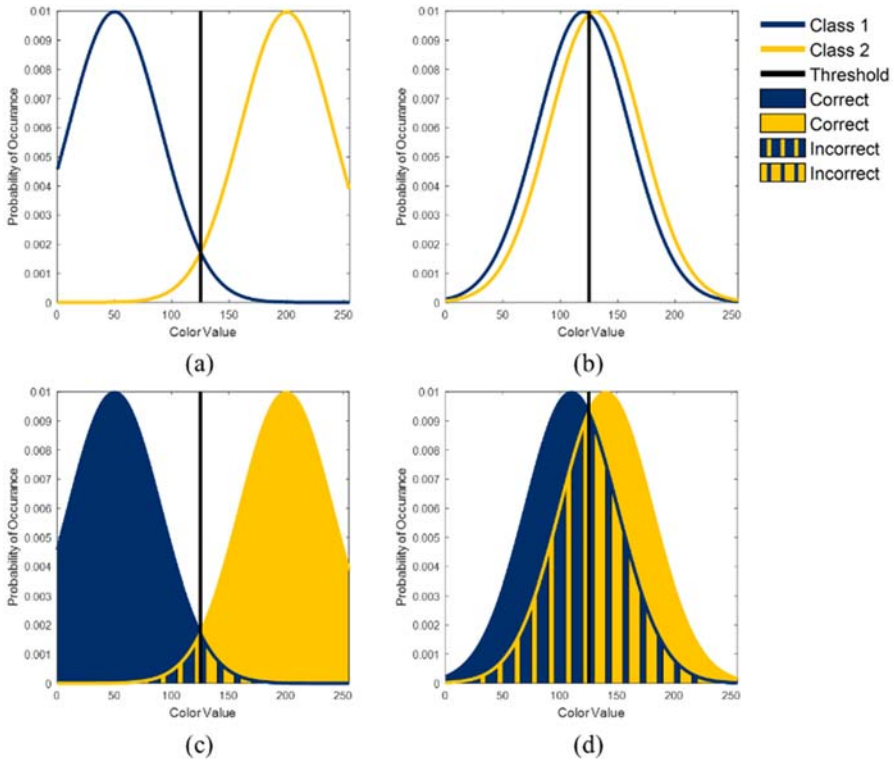


Figure 21. Two distributions that need to be segmented are shown in (a) and (b). A threshold is set and the same two distributions separated as shown in (c) and (d) where solid sections show the proper classification and hatched section show improper classification.

2.3.7 Image Stitching

Image stitching or mosaicking is a technique that is used to put together individual images to create a larger image [149]. One method that is used on many cellular phones is

the panorama. This technique relies on the focal point of the camera remaining in the same location while all of the images are taken. If this condition is not satisfied, parallax occurs within the image causing discontinuities or the panorama will fail [137]. Parallax is the case where two images are stitched together and the background appears to have doubles of artifacts in it. An example of parallax is shown in Figure 22. The green boxes show the steel grid that was stitched together properly. The yellow and red boxes highlight the parallax caused in Figure 22 (c).

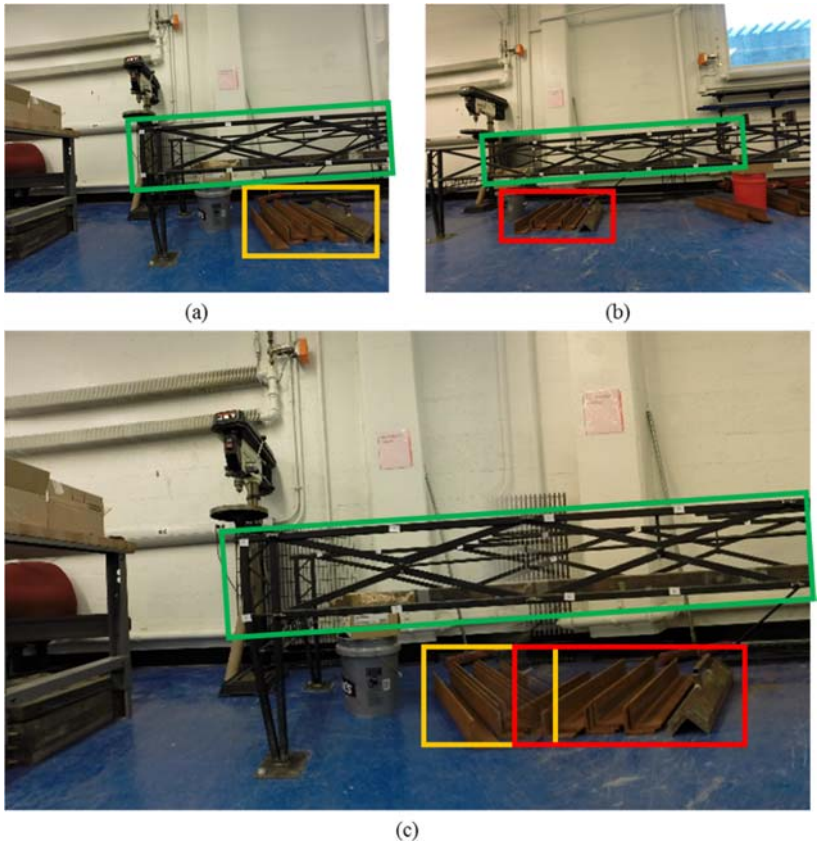


Figure 22. (a) Image 1. (b) Image 2. (c) Stitched Image.

It is clear that the background boxed in yellow and red in Figure 22 (c) are the same object. They appear twice in the stitched image because of the two different angles the image was taken from. Instead of being rotated like what is done in a panorama, the images

camera translated while these images were taken. The object of interest for stitching was stitched properly, but all of the other distances that were not in the same plane were stitched improperly. There are several methods that can be used to match points within images. The simplest method is to manually identify the correspondences. This is a laborious and tedious process, so more automated methods such as corner detectors, scale invariant feature transform, and many others [138].

2.3.8 Civil Applications

The image processing techniques described in this section are just a few of the methods that exist for enhancing the output of images and segmentation and have been leveraged for many applications, including civil applications. Crack identification algorithms leveraging these and other techniques is important to bridges, buildings, dams, and many other forms of infrastructure. Matched filtering [150], percolation [130], edge detection [87], and several other algorithms [40] have been leveraged to identify this form of damage. Texture was used to monitor corrosion growth [151] and other forms of related damage [99]. Furthermore, global views of structures have been created and mapped for better viewing [109, 152]. A good review of what different image processing and computer vision methods that have been used for concrete civil structures can be found in [40].

2.4 Camera Sensors, Geometry, and Measurement Algorithms

2.4.1 Sensors and Camera Inputs

There are two major types of camera sensors that are easily accessible: charged couple devices (CCD) and complementary metal–oxide–semiconductor (CMOS) [153]. In general, CMOS sensors are more common because they process data faster and are less expensive than CCD sensors. CCD sensors historically have been a better sensor because

they have lower noise, a global shutter, and perform better in low light conditions. CMOS sensor technology has improved significantly and are now more comparable to CCD sensors, but are easier to manufacture. In most measurement systems, CCD sensors are still used due to the low noise associated with the sensors.

Another important parameter that is determined by the camera sensor is the way the image data is saved. Monochrome cameras do not have color associated with them and achieve high spatial resolution in the band of light the sensor accepts. Other sensors have a spacing that will collect different bands of light with different detectors throughout the sensor. An example of this type of sensor is shown in Figure 23 which shows a Bayer filter with red, green, and blue detectors [154].

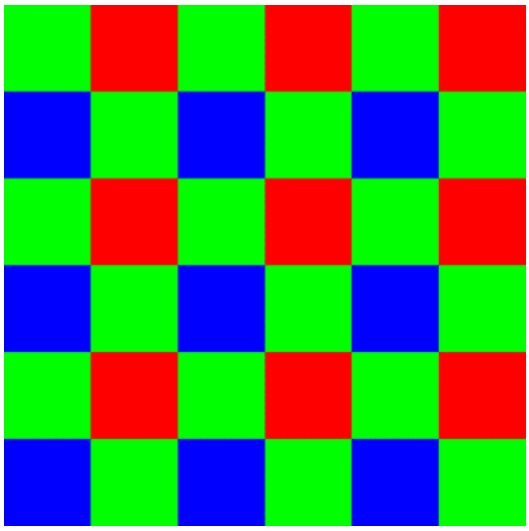


Figure 23. Bayer filter detector pattern [154]

In this case, the image has three color bands, one for Red, one for Green, and one for Blue (RGB). From the figure, it is clear that the green band has the highest spatial content. This sensor has an output in the RGB color space, but this is not the only color

space a sensor can output or that the color space can be converted into [118, 154]. Other color spaces include Cyan, Magenta, Yellow, and blacK (CMYK) and the Hue Saturation and Intensity (HSI) [118]. In general, RGB color space operations are more difficult than color spaces like HSI that separate the color from the intensity. This is because changes in the RGB color space values result in large changes in the visual appearance of the color [118]. An example of this can be clearly seen in Figure 24 and Figure 25. The top left image is the same red color labeled original image. Each of the other images in the figures are 50% changes in each of the color bands. It is clear from the results that the HSI color space keeps the visual appearance of the color of the original image much better than the RGB color space. It is for this reason that in histogram equalization across color images, the RGB space should not be used and the histogram equalization should be used on the intensity band of the HSI color space [118].

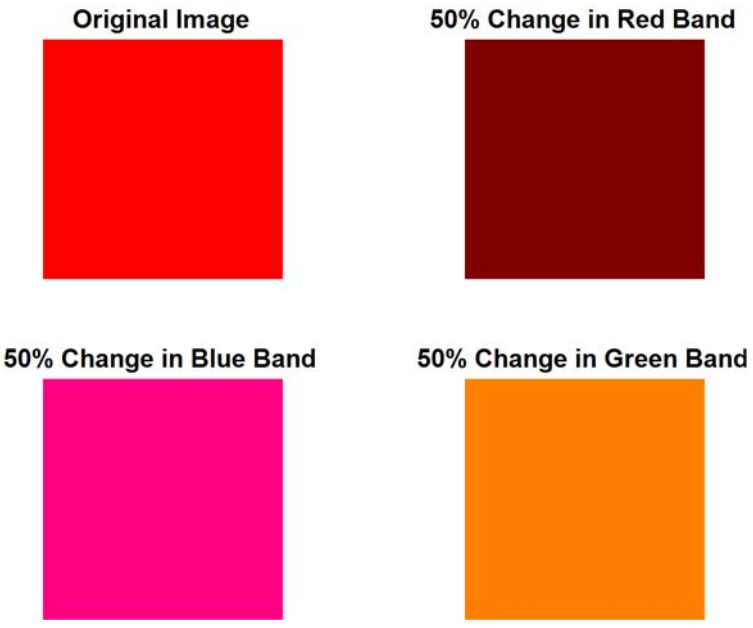


Figure 24. Adjusted values in the RGB color space.

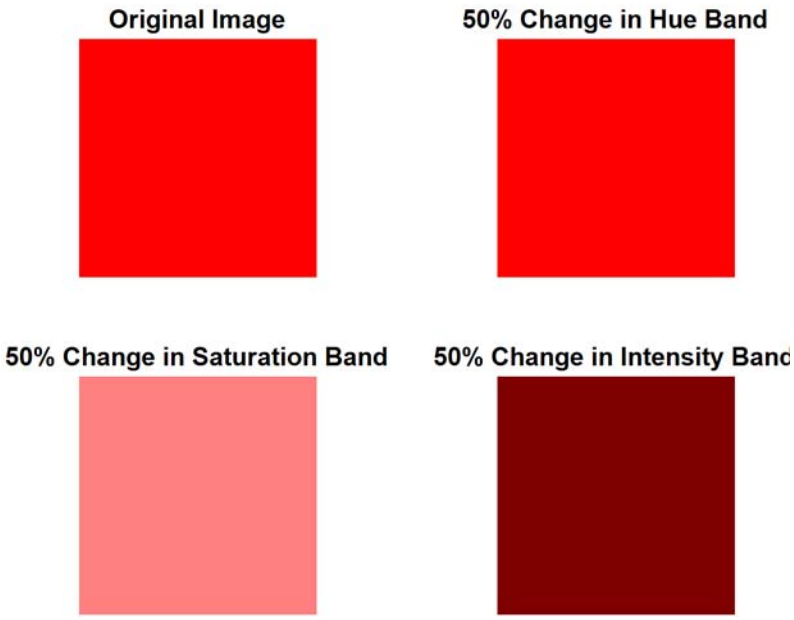


Figure 25. Adjusted values in the HSI color space.

The camera parameters that determine the quality of the image are: ISO, shutter speed, focal length, aperture, and target distance. The ISO setting determines how sensitive the sensor is to light. Higher values of ISO correspond to higher sensitivity, but the higher sensitivity also increases the amount of noise in the image. The shutter speed determines how fast the image is acquired. At high shutter speeds, the images are taken quickly, but as a result, less light passes through to the sensor. Aperture also controls the light allowed to pass through to the sensor. Smaller openings allow less light to pass, but increases the distance in the scene that is in focus or depth of field. The target distance determines the size of the pixels in the scene. This is important for metrology. As distance increases, the physical size of the pixels in the scene increase [155].

2.4.2 Coordinate Systems

There are four coordinate systems in an image shown in Figure 9. The pixel coordinates x_p and y_p (red) are the default coordinates in pixels in the image. The image

coordinates x_i and y_i (purple) are the pixel coordinates shifted to the center of the image. The camera centered coordinates x_c, y_c, and z_c (black) are three dimensional coordinates with the origin at the optical center. The world coordinates x_w, y_w, and z_w (green) are the actual coordinates of objects in a three dimensional scene with a predefined origin. The pixel coordinates and image coordinates are in a two dimensional coordinate system measured in pixels. The camera coordinates are three dimensional scaled coordinates where the relative positions of the points are correct, but the physical dimensions are not assigned. Finally, world coordinates are three dimensional that have physical dimensions based on a scale bar within the image. The origin can be arbitrary or assigned by the user depending on the application.

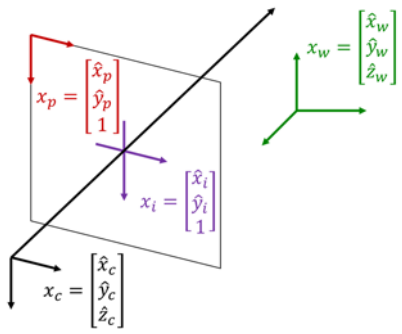


Figure 26. Coordinate systems [156]

The coordinate systems displayed in Figure 26 are crucial to the concepts presented in sections 2.4.4, 2.4.5, 2.4.6, and 2.4.7. These sections focus on camera geometry which relies heavily on converting between these coordinate systems as well as identifying the transformations between multiple camera views of a scene.

2.4.3 Camera Calibration

All cameras lenses cause some distortion due to the bending of light into the sensor. Wide angle lenses suffer more from this problem because the light has to bend significantly

more to reach the image plane at the edges of the image plane. Figure 27 shows the light rays and field of view of two different camera systems. The red camera system has a shorter focal length and a larger field of view. The blue camera system has a longer focal length and narrower field of view. The red rays that approach the edges of the lens must bend significantly more to hit the edges of the image plane than the light rays of the blue camera system.

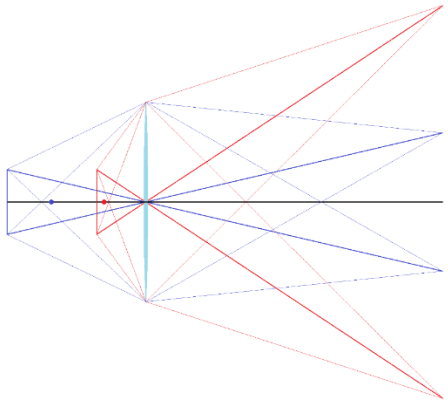


Figure 27. Light rays for two different focal lengths [157]

The multiplane camera calibration technique [158] was used to fit an equation to the distortion to correct it. The camera distortion was modeled for tangential, radial, and skew distortion. The corrected coordinates are shown in equations (2.17) and (2.18) [137, 159]. The x and y are the normalized image projection coordinates based on the pinhole projection model, $r = \sqrt{x^2 + y^2}$, k_{ri} is the i^{th} radial distortion coefficient to be estimated, and k_{ti} is the i^{th} tangential distortion coefficient to be estimated [159]

$$x_d = \left(1 + k_{r_1} r^2 + k_{r_2} r^4 + k_{r_3} r^6\right) x + 2k_{t_1} xy + k_{t_2}\left(r^2 + 2x^2\right) \tag{2.17}$$

$$y_d = \left(1 + k_{r_1} r^2 + k_{r_2} r^4 + k_{r_3} r^6\right) y + k_{t_1}\left(r^2 + 2y^2\right) + 2k_{t_2} xy \tag{2.18}$$

The technique used the known locations of the intersections of the checkerboard to be the reference points to correct the image. The more images taken from different locations, the better the approximation is expected to become. The algorithm was implemented in MATLAB using the toolbox developed by Bouget [159] and the MATLAB Camera Calibrator App [160, 161]. In both cases, multiple images of a checkerboard were used to identify corners, minimize error in the equations above based on the known points and estimate the parameters of the radial and tangential distortion which could be used to correct the distortion in other images with the same camera parameters. However if a camera with an adjustable lens is used, every time the lens parameters are changed, the camera must be calibrated again to correct for lens distortion.

2.4.4 Homography

Another technique that was used to complete much of this work is homography, which is used to project the image onto a plane in that image. The homography matrix is a transformation matrix that converts the existing pixel coordinates (x, y) in the image to the desired coordinates (X, Y) [143].

$$\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{2.19}$$

The coordinates (X, Y) and (x, y) have an added 1 at the end which serves as a placeholder for the third (out-of-plane) dimension. The three dimensional scene is projected onto a two dimensional plane when the image is captured. This projection onto the image plane effectively eliminates the out of plane dimension. Therefore, the third dimension added back to the original coordinates and the desired coordinates artificially.

Since both artificial values are 1, the scale of the image does not change so the distance to the target would also not change if the three dimensions were still available.

The homography matrix is calculated using at least four points known points in the image that are in the same plane. Each known point in the image (X, Y) corresponding to the image point in pixels (x, y) provides two equations in the system of equations as shown in the equation below (2.20) [162]. The system of equation is only composed of 8 equations if only four points are used and there are 9 variables associated with it. The last variable can be set to 1 because the transformation does not have scale.

$$[A]^{[2n \times 9]}[h]^{[9\times 1]} = [0]^{[2n \times 1]}$$

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1X_1 & -y_1Y_1 & -X_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2X_2 & -y_2Y_2 & -X_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & 1 & 0 & 0 & 0 & -x_nX_n & -y_nY_n & -X_n \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1Y_1 & -y_1X & -Y_1 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2Y_2 & -y_2X_2 & -Y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & x_n & y_n & 1 & -x_nY_n & -y_nX_n & -Y_n \end{bmatrix} \times \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \\ h_9 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (2.20)$$

If more than 4 points in the plane are used, the system of equations is over determined, but still lacks scale. In this case, the homography can be calculated using a least squares approach shown in equation (2.21).

$$(A \times h)^T \times (A \times h) = h^T \times (A^T \times A) \times h = 0 \quad (2.21)$$

The minimization can be determined using the eigenvector with the smallest eigenvalue of the A^TA matrix. The values in the eigenvector are then rearranged to become the homography matrix.

$$eig\left(A^T A\right) \rightarrow E_{vec}\left(\min\left(E_{val}\right)\right) \rightarrow \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \\ e_8 \\ e_9 \end{bmatrix} = \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix} \tag{2.22}$$

The inverse mapping of the homography matrix can be used to map all of the pixel colors onto the new plane, but since the homography matrix has no scale, the scale must be set by the user. If the user defines a scale that is too large, edge localization is difficult because of artificial blur due to low resolution, whereas if the scale is set to be too small, the resolution will not be high enough in the final image to obtain good measurements. The resulting image from this step can heavily affect the accuracy of the measurement. To circumvent this issue, the image scale is set to achieve a new image that is of similar size to the original image (usually slightly larger) to ensure no resolution was lost. Backward mapping and bilinear interpolation were utilized to ensure that all of the pixels in the new image had color associated with them. If forward mapping were used, there would be black pixels in the new image which would also make measurements difficult. An example of the results of the transformation on an image is shown in Figure 28 where the four corners of the paper were used to project the image onto the plane in the image.

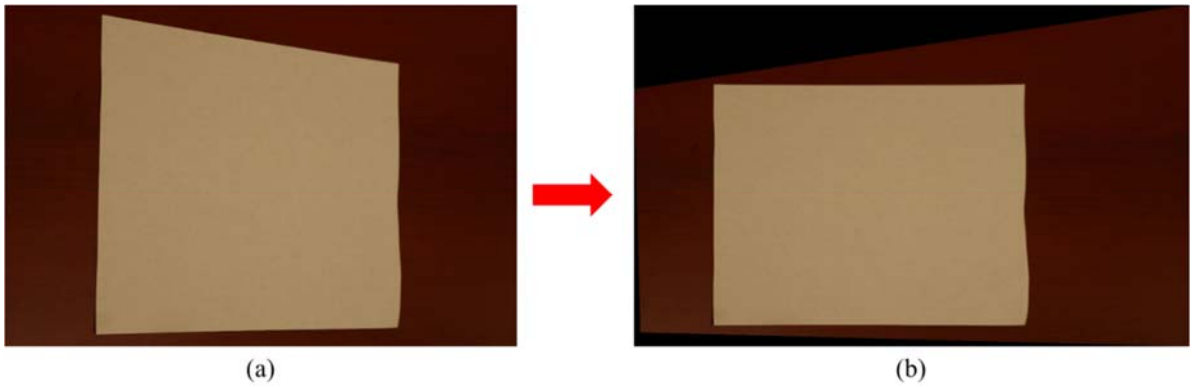


Figure 28. Homography transformation: (a) Original image (b) Image flattened to the plane of the paper.

The angle of the camera with respect to the measurement plane will also affect measurement accuracy due to interpolation and errors in the known point selection. Larger angles create large errors in pixels that encompass a larger area in the corrected image. Also, measurements must only be taken in the plane the image is projected onto. Once the image is corrected for distortion and projected onto the measurement plane, measurements can provide a better estimate to quantify the size of the damage in the images in that plane.

2.4.5 Photogrammetry

This approach leverages fiducial points in the images to obtain high accuracy 3D position of those points. The setup is generally easier than DIC because the speckle pattern is not required. Many researchers have worked with photogrammetry for civil infrastructure applications [104, 163-166].

The computation of three dimensional points in images relies on triangulation [167]. Figure 29 shows a picture of why two images are required to obtain the three dimensional location of a point in space.

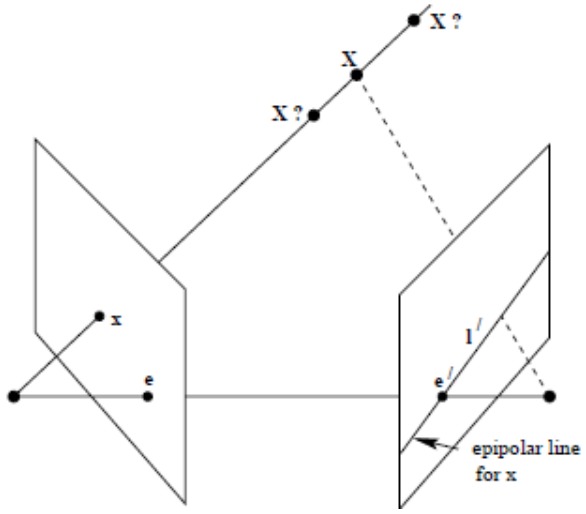


Figure 29. Triangulation of point x using two images [143].

The major problem with using a single camera is that the rotation and translation are not known between two images. The fundamental matrix (F) is used to relate the two images and calculate the rotation and translation. There are several methods used to calculate the fundamental matrix with different numbers of corresponding points [143], however, all must satisfy equation (2.23).

$$x_{p_1} \times F \times x_{p_2} = 0 \tag{2.23}$$

where x_p are corresponding points in the images and F is the 3x3 fundamental matrix. At least 8 points are needed to solve for F. The essential matrix can be computed from the fundamental matrix using equation (2.24) [143] and separated into relative camera rotation and translation using equation (2.25) [168]. K and K' are the camera intrinsic matrices of the two cameras which are generally defined by equation (2.26) where f is the focal length in pixels, gamma is the skew which is usually zero or very close to zero, and C_x and C_y are the center location of the image in pixels. This matrix is an output of the

camera calibration explained in section 2.4.3. If using the same camera, the K matrices are the same.

$$E = K' \times F \times K \tag{2.24}$$

$$E = [t]_x \times R \tag{2.25}$$

$$K = \begin{bmatrix} f & \gamma & C_x \\ 0 & f & C_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.26}$$

Using singular value decomposition, the rotation and translation can be calculated.

$$\begin{bmatrix} U & S & V \end{bmatrix} = svd(E) \tag{2.27}$$

After decomposing E, S is normalized such that the first two diagonal values are 1 and the rest of the matrix is 0. W is set to be an orthogonal matrix with the last element equal to 1. Using this and equations (2.25) and (2.27), the translation and rotation can be calculated using equation (2.28).

$$E = USV^T = U\left(WSU^T UW^{-1}\right)V^T = \left(UWSU^T\right)\left(UW^{-1}V^T\right) = [t]_x R \tag{2.28}$$

The translation is represented at a dual matrix which can be changed into a vector using equation (2.29).

$$\begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \rightarrow \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} = T = [t]_x \tag{2.29}$$

After calculating the rotation and translation, there are 4 solutions since the essential matrix is not of full rank, R and t, R and –t, R^{-1} and t, and R^{-1} and –t [143]. There is only one solution that will allow the points in space to be in the field of view of both cameras which is the solution that is correct.

After calculating the rotation and translation, the projection matrices of each image are used to triangulate the three dimensional camera centered points [167]. Equations (2.30) and (2.31) show the projection matrices for each camera. The first is set to have zero rotation and translation and the second is rotated and translated according to the decomposition of the essential matrix.

$$P = \begin{bmatrix} K & 0 \end{bmatrix} \tag{2.30}$$

$$P' = \begin{bmatrix} K'R & K't \end{bmatrix} \tag{2.31}$$

Using the known corresponding points, the three dimensional points were calculated using equation (2.32) where P(i, :) represents the i'th row of the projection matrix of each image. The solution to Ax=0 is determined using singular value decomposition. The vector corresponding to the smallest singular value is then rearranged to determine the scaled camera coordinates [167].

$$\begin{bmatrix} x_p P(3,:) - P(1,:) \\ y_p P(3,:) - P(2,:) \\ x_p' P'(3,:) - P'(1,:) \\ y_p' P'(3,:) - P'(2,:) \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = 0 \tag{2.32}$$

After the camera three dimensional scaled coordinates are calculated, the coordinates must be converted into world coordinates [169]. Using the known positions of fixed points in the image, a direct linear transformation can be used to convert the camera coordinates to world coordinates using equation (2.33). This equation is very important because if certain coordinates are known, they can be fixed to make the resulting transformation more accurate.

$$\begin{bmatrix} x_1 & y_1 & z_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -x_{G1}x_1 & -x_{G1}y_1 & -x_{G1}z_1 & -x_{G1} \\ 0 & 0 & 0 & 0 & x_1 & y_1 & z_1 & 1 & 0 & 0 & 0 & 0 & -y_{G1}x_1 & -y_{G1}y_1 & -y_{G1}z_1 & -y_{G1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x_1 & y_1 & z_1 & 1 & -z_{G1}x_1 & -zy_1 & -z_{G1}z_1 & -z_{G1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & z_n & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -x_{Gn}x_n & -x_{Gn}y_n & -x_{Gn}z_n & -x_{Gn} \\ 0 & 0 & 0 & 0 & x_n & y_n & z_n & 1 & 0 & 0 & 0 & 0 & -y_{Gn}x_n & -y_{Gn}y_n & -y_{Gn}z_n & -y_{Gn} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x_n & y_n & z_n & 1 & -z_{Gn}x_n & -z_{Gn}y_n & -z_{Gn}z_n & -z_{Gn} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \\ T_7 \\ T_8 \\ T_9 \\ T_{10} \\ T_{11} \\ T_{12} \\ T_{13} \\ T_{14} \\ T_{15} \\ T_{16} \end{bmatrix} = 0 \quad (2.33)$$

Singular value decomposition is used to solve this system of equations and the last vector of the V matrix corresponding to the smallest singular value is used to calculate the transformation shown in equation (2.34).

$$T = \begin{bmatrix} T_1 & T_2 & T_3 & T_4 \\ T_5 & T_6 & T_7 & T_8 \\ T_9 & T_{10} & T_{11} & T_{12} \\ T_{13} & T_{14} & T_{15} & T_{16} \end{bmatrix} \quad (2.34)$$

Point based photogrammetry methods have been demonstrated for many applications [170]. For instance, point based methods were applied a beam in laboratory conditions [14, 165]. In addition, photogrammetry was applied to track the movement of a wind turbine using high speed stereo cameras [171]. Furthermore, these methods have been extended to small span bridge applications [163, 164]. Malesa et al. used a target placed on a railway bridge to measure deformation during a train was crossing [103]. Furthermore, commercial photogrammetry systems have been developed to calculate movement and measure deformation leveraging coded markers, uncoded markers, and calibrated scale bars [172, 173]. More recently, a photogrammetry method was extended to UAS to measure beam deflection [174].

2.4.6 Structure from Motion

Structure from motion (SFM) uses the movement of a single camera to estimate the three dimensional locations of keypoints in a set of images to output a point cloud. There are many ways to extract the keypoints in the image, but one of the more popular methods is the SIFT algorithm [138]. The algorithm uses a multiresolution approach to obtain a 128 element feature vector for each point that is a local extrema which is used to match points between images. Then, a random sample consensus (RANSAC) algorithm is used to identify the transformation between the images [175]. The three dimensional points are then extracted and an error minimization is used to produce the final point cloud. An example of the results of SFM is shown in Figure 30 on a bridge pier. For this case, over 400 images of the pier were collected similar to the ones shown in Figure 30. The images were corrected for lens distortion and using VisualSFM [176], SIFT, RANSAC, and the bundle adjustment were computed and the 3D points were extracted.

Figure 30. Sample images and results of SFM

The use of point clouds is generally for augmented reality purposes and model generation [30, 37, 39, 177]. The difficulties in obtaining measurements from structure from motion is the accuracy of the point cloud; however, there is current research in obtaining high accuracy point clouds for deformation measurements using a hierarchical approach [28, 29].

2.4.7 Digital Image Correlation (DIC)

Optical flow can be considered the simplest form of digital image correlation (DIC). Both are used to track movement in images through light intensity changes. It is possible to obtain the movement using natural texture, but a random speckle pattern dramatically can improve results. The Lucas-Kanade algorithm for optical flow is one of the more popular algorithms [178]. This is a limited form of digital image correlation that only takes into account rigid transformations of the subsets of pixels in the image and subpixel movement. In cases where the movement is more than one pixel, a multiresolution approach can be used to iteratively force subpixel movement and update the movement to obtain the full movement. Equation (2.35) shows the comparison of an image and the shifted image. H is the original image and I is the image moved by u and v pixels in the horizontal and vertical directions respectively and x and y are the position of the pixels in the image.

$$I\left(x+u, y+v\right) = H\left(x, \mathrm{y}\right) \tag{2.35}$$

The moved image can be approximated by

$$I\left(x+u, y+v\right) \cong I\left(x, y\right) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + HOT \approx I\left(x, y\right) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v \tag{2.36}$$

Combining equation (2.35) and equation (2.36),

$$I(x,y)+\left(\frac{\partial I}{\partial x}u+\frac{\partial I}{\partial y}v\right)=H(x,y) \tag{2.37}$$

Rearranging,

$$-\left(I(x,y)-H(x,y)\right)=\begin{bmatrix}\frac{\partial I}{\partial x} & \frac{\partial I}{\partial y}\end{bmatrix}\begin{bmatrix}u\\ v\end{bmatrix} \tag{2.38}$$

Using these equations, small movements in image subsets can be extracted using different methods such as Newton-Raphson [179] and an example of the results of the algorithm is shown in Figure 31. The gear was rotating when the two images were taken. The smaller gear was rotating counterclockwise and the larger gear was rotating clockwise. The red arrows shown in the image and highlighted in the red box indicate the movement of the gear.

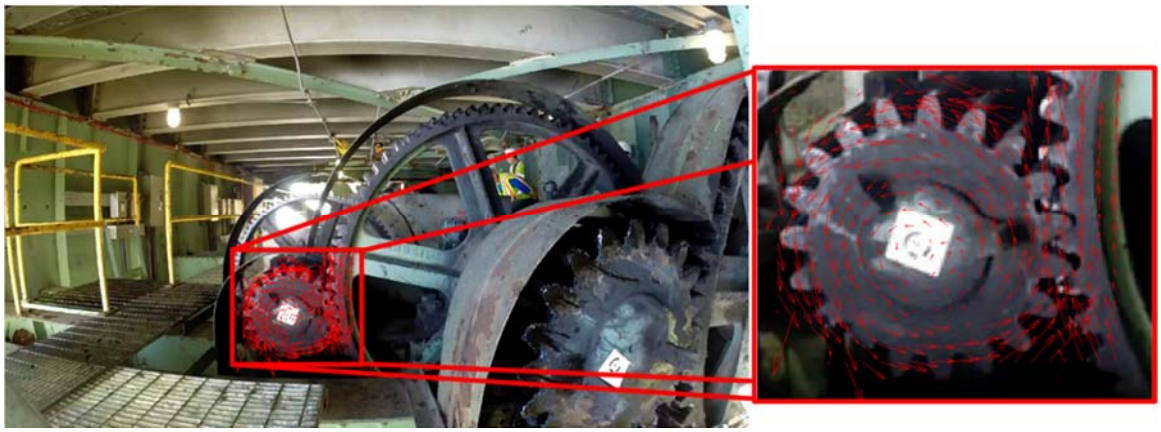


Figure 31. Results of optical flow for a rotating gear

In many cases, optical flow cannot be used because more complex motion must be characterized. A more complex DIC algorithm is needed to complete these tasks. In commercial software used for DIC, a speckle pattern is used to add artificial texture to the images to make the tracking more accurate. Instead of using rigid body motion, DIC can

use affine or higher terms to track the changes in light intensity of each subset which allows for more complicated changes to be tracked [180]. DIC is scale independent and it has been used for many applications from the micro-scale to the structural-scale [18, 181, 182]. This method has been extensively applied to experimental testing to the full field outputs of deformation, strains, and stresses in the material. Two dimensional DIC uses a single camera and has been extensively studied [179]. If multiple cameras are used, a third dimension can be extracted and measured using triangulation. Several commercial systems have been developed to perform 2D and 3D measurements with DIC including ARAMIS by GOM, VIC 3D by Correlated Solutions, and DaVis by LaVision. 2D DIC can be performed with a single camera while 3D DIC requires at least two cameras and triangulation for reasons shown in Figure 29.

One of the challenges associated with DIC is the application and quality of the speckle pattern. Application of the pattern can be as simple as spray painting the pattern with a stencil or spraying over the sample and allowing the paint to fall randomly on the specimen [182, 183]. Other methods can be more difficult such as applying particles to a specimen [181]. Different techniques have different variability making it difficult to determine if a pattern is good before testing. Therefore, several metrics were developed to determine the quality of a speckle pattern within an image to help determine the systematic and random error in measurements [179, 184-187]. The two general classes of DIC speckle pattern quality are local and global metrics. Local metrics compare features of the speckles within the subset or facet of the image. These methods then compare the facet values to obtain the quality of the pattern. Global metrics are based on all of the pixels in the images with some feature of the pixels. Some of the features DIC pattern metrics can rely on are

intensity gradients, color contrast, and entropy of the pixels. All of these metrics follow the process of identifying the features at every pixel and averaging them across the facet in local metrics and the image in global metrics to determine a single value to compare the patterns. Global metrics are more convenient because the metric breaks down to one number. The local metrics break down to several numbers corresponding to each facet. Mean Intensity Gradient (MIG) is defined in equation (2.39) where f_x gradient in the x direction, f_y gradient in the y direction, M is the width of the image or subset, and N is the height of the image or subset [187].

$$MIG = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\sqrt{f_x(i,\mathrm{j})^2 + f_y(i,j)^2}}{MN} \tag{2.39}$$

The sum of square of subset intensity gradient is shown in equations (2.40) and (2.41) where the variables are the same as equation (2.39) [187].

$$SSSIG_x = \sum_{i=1}^{N}\sum_{j=1}^{M} f_x(i,j) \tag{2.40}$$

$$SSSIG_y = \sum_{i=1}^{N}\sum_{j=1}^{M} f_y(i,j) \tag{2.41}$$

The entropy metric can be calculated using equation (2.42) where I_P is the intensity of the current pixel, I_i is the intensity of each neighboring pixel, n is the number of neighbors, and M is the width of the image or subset, and N is the height of the image or subset [187].

$$Entropy = \frac{\sum_{P\in S}\sum_{i=1}^{n}\left|I_P - I_i\right|}{2^{\beta} MN} \tag{2.42}$$

In all of the metrics described, the M and N can be either the image size or the subset size for the global or local metric respectively. Though the quality of the speckle pattern is very important parameter that determines the quality of the image measurements, the facet size and spacing used in the DIC software are also very important to the performance of the algorithm. The facet or subset size should be able to encompass at least 3 speckles. The speckles should be between 3 to 5 pixels in size to achieve good measurements of displacement. The facet size is the size of the neighborhood around each measurement point. Large facet sizes result in higher accuracy, but also average out the localized displacements. Smaller facets achieve better local measurements, but tend to add more noise to the displacement field. Therefore, when local displacement measurements are required, the smallest facet size possible is desirable, but when local displacement measurements are not as important, larger facets may be desirable. The spacing is how far away each facet center is from the neighboring facet center. The spacing is important because too much facet overlap results in high computation time and too little overlap results in poor measurement quality. A diagram of facet overlap and size is shown in Figure 32.

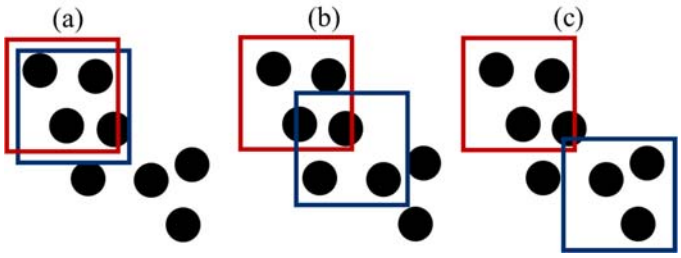


Figure 32. Two adjacent facets with at least 3 speckles and (a) 90% overlap, (b) 50% overlap, and (c) 10% overlap.

Each facet is tracked individually with a DIC algorithm. There are many ways to perform DIC and different metrics to determine how correlated a facet is in the new image. In addition, the choice of a shape function will also determine the performance of the algorithm. In general, the commercial DIC systems do not provide the shape functions used, but they generally use high order shape functions for high quality measurements [180]. The shape functions determine the quality of the measurements between the facet center points. In Figure 33, $P(x_0, y_0)$ is the point in which the measurement for the facet deformation is taken [179, 180, 184]. Point $Q(x_i, y_i)$ must be interpolated between neighboring facets using a shape function [179, 180, 184]. If one facet was used for every pixel in the image, the interpolation would only be required for locations between pixels, but since this is computationally inefficient and it is expected that neighboring pixels move in the same direction, subset overlap is usually set to between 40% and 60%. This ensures that enough measurement points are calculated to achieve good full field displacements, while still being computationally practical.

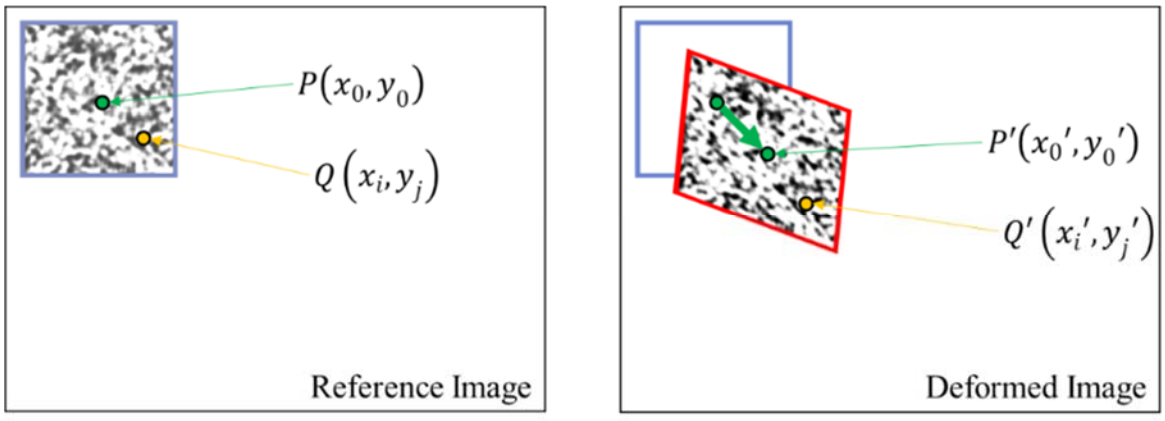


Figure 33. Undeforemed and Deformed subset with two corresponding points [179, 180, 184].

2.4.8 Civil Applications

Photogrammetry has been used to measure structural deformation in lab scale experiments as well as field experiments [103, 188]. It has been used to measure deformation of wind turbine blades and many other applications [170]. Full field DIC techniques have been applied to walls and buildings to perform experiments simulating an earthquake [18, 182]. DIC has also been used to obtain deformation measurements of railroad ties under loading [189]. Structure from motion has been used to model bridges and buildings [28, 92, 190]. The noncontact nature of these methods has major advantages compared to contact systems which could make them more attractive for use in the field in the future.

2.5 Multispectral and Hyperspectral Imagery

2.5.1 The Electromagnetic Spectrum

Multispectral imagery is a subset of light ranges that are collected by imaging sensors to provide multiple channels within the image. For instance, a red, green, and blue (RGB) image obtained from a standard camera is a multispectral camera with three channels. Multispectral cameras can have many channels at different wavelengths of light. One of the challenges associated with multispectral imagery is that features in one band can appear in the other bands as well which can make identification of objects within the images difficult. Hyperspectral imaging is different because it collects many channels with very small, targeted ranges light wavelengths that do not overlap [191]. This eliminates the problem associated with crosstalk between bands of the multispectral imagery, but requires many bands which make the images more difficult to capture and require higher amounts of storage for the data. Many platforms have been used for hyperspectral imagery with

varying resolution and mobility [192]. Figure 34 shows the electromagnetic spectrum and highlights how small the range of wavelength is for visible light.

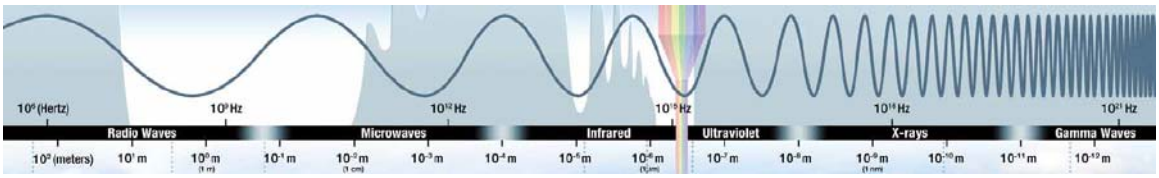


Figure 34. Electromagnetic Spectrum [193]

2.5.2 Resolving Power

The angular resolution determines the minimum power of the optics to resolve a feature in the image. The Airy disk is defined as the first dark ring in the Fraunhofer diffraction pattern [194]. The Airy disk is determined by equation (2.43) [194] where $J_1(x)$ is the first-order Bessel function, θ_P is the angle, λ is the wavelength, I_0 is the intensity at the center of the disk, and a is the size of the aperture.

$$I\left(\theta_p\right)=4I_0\left(\frac{J_1\left(\frac{2\pi}{\lambda}\cdot a\cdot\sin\left(\theta_p\right)\right)}{\frac{2\pi}{\lambda}\cdot a\cdot\sin\left(\theta_p\right)}\right) \tag{2.43}$$

Diffraction only becomes a concern for very small apertures or very small image sensors resulting in small pixel sizes. The optimal aperture diameter (D) can be calculated using equation (2.44) where c is a constant, f' is the effective focal length and λ is the wavelength [195].

$$D=c\sqrt{f'\lambda} \tag{2.44}$$

When the optimal diameter is compared to the actual diameters used to capture images, the resolution is generally not limited by diffraction for visible light. Most of the time, blur is related to the camera parameters and focus of the image planes. For longer focal lengths however, the airy disk can cause blur due to diffraction for small apertures or long wavelengths. The airy disk equation is shown in equation (2.45) [194] where f' is the effective focal length, λ is the wavelength, D is the aperture diameter and d is the diameter of the airy disk.

$$\frac{d}{2} = 1.22\frac{f'\lambda}{D} \tag{2.45}$$

When the diameter of the airy disk is the same physical dimension of the pixel on the image sensor, additional pixels will not provide higher resolution. Figure 35 shows the size of the airy disk for different wavelengths calculated using equation (2.43).

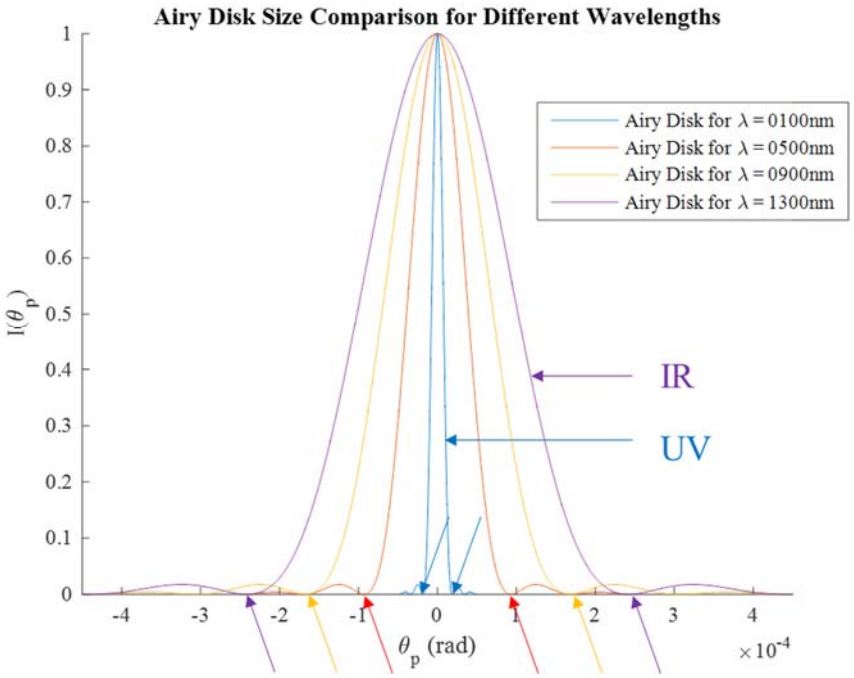


Figure 35. Airy disk size for different wavelengths of light calculated from equation (2.43)

The size of the disk is much larger for longer wavelengths showing that their power to resolve objects is lower than that of shorter wavelengths. In some cases, resolution is not the only reason for using light of shorter wavelength such as measurement in high temperature applications, but it can also be used to increase resolution of the camera system if the system is in a diffraction limited case [194, 196]. A better illustration of how smaller wavelengths can achieve higher resolving power is shown in Figure 36. The distance from the center of each peak is 50 micrometers and the wavelengths are 400nm and 750nm. The blue and red dots show where the airy disks cross for each wavelength. It is clear that there is better separation for the shorter wavelength showing that it has better resolving power.

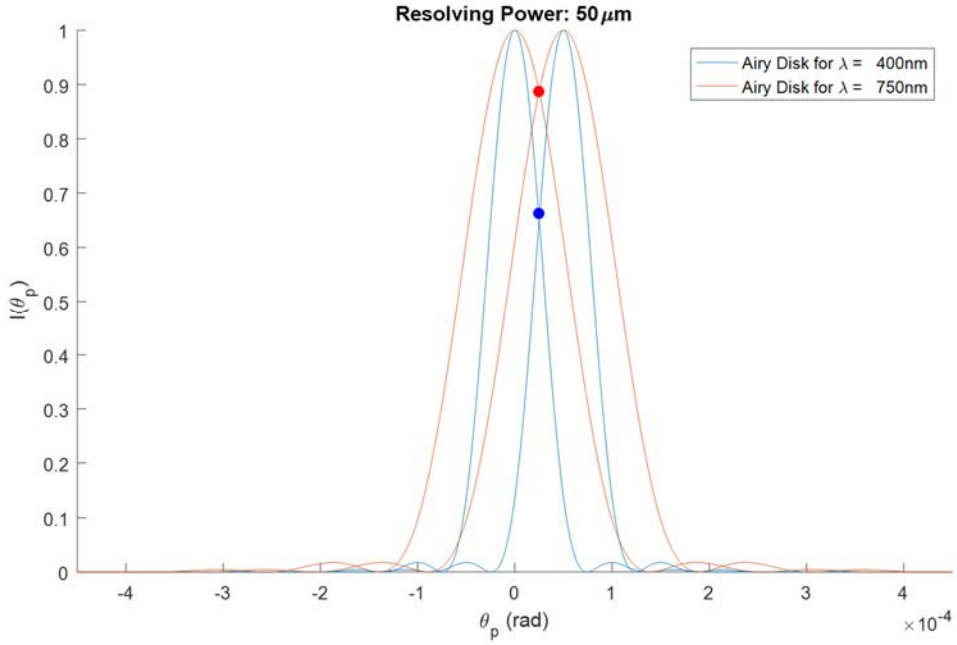


Figure 36. Resolving for opposite ends of the visual spectrum for UV in blue and IR in red.

2.5.3 Light Filters

Another sensor parameter that is vital to the output of the sensor is the choice of light filter and the choice of monochrome and color sensors. Light filters are usually put

into place to eliminate the light outside the visible range before it reaches the sensor [197]. The glass lens also can serve as a filter for ultraviolet light [198]. The different filters can be used to produce different imaging effects that can be used for many different applications. The sensors themselves are more sensitive to different bands of light and if the filter and sensor are tuned to a specific band of wavelengths, the results are generally better. It is possible to convert a standard color camera to allow near infrared light to appear, however it is not possible to convert the camera into an infrared camera because the visual light sensors are not sensitive to infrared light [197].

2.5.4 Infrared Thermography

Infrared thermography can capture temperature data (also known as radiometric data) using infrared light which is emitted from the surface. Infrared thermography has been leveraged for a number of applications from health and medicine to maintenance and inspection [199-201]. One important parameter that effects this method significantly is emissivity (ε(T)), defined in equation (2.46), which is the ratio radiation emitted by a material (E(T)) and the radiation emitted by a blackbody at the same temperature ($E_b(T)$) where T is the temperature in Kelvin [202].

$$\varepsilon(T) = \frac{E(T)}{E_b(T)} \tag{2.46}$$

The power emitted from a blackbody is defined by the Stefan-Boltzmann law shown in equation (2.47) where σ is the Stefan-Boltzmann constant ($5.67*10^{-8}$ Wm^2K^{-4}) [202].

$$W_b(T) = \sigma T^4 \tag{2.47}$$

Planck's law is defined below as a distribution of emitted power shown in equation (2.48) where $C_1=3.742*10^8$ W* $\mu m^4/m^2$ and $C_2=1.439*10^4$ μm*K [199, 202].

$$W_{\lambda,b}(\lambda,T)=\frac{C_1}{\lambda^5\left(e^{\frac{C_2}{\lambda T}}-1\right)} \tag{2.48}$$

Infrared sensors that are used in infrared cameras measure the amount of infrared photons that hit the detector which is effectively a measurement of the radiation emitted by the material. This can then be extrapolated to determine the temperature in the scene. The problem with this approach is the different emissivity values. If the emissivity of the material is well known, this is not a problem, but in many cases, the emissivity is unknown which can cause errors in the temperature measurement. Some infrared cameras do not do this process and result in images of the infrared spectrum which are scaled to appear nicely in the image using some of the techniques used in section (2.3). Infrared cameras that do use this process result in radiometric data which correspond to temperature measurements of the scene. This information is useful for many applications [199].

2.5.5 Civil Applications

Multispectral imagery outside of the visual spectrum has been limited to mainly infrared use due to the ability to convert the response on the sensor to temperature measurements. Infrared thermography has been used to identify grouted areas in a masonry wall [203]. Passive infrared imagery has also been used to identify delaminations in a bridge deck [204, 205]. Furthermore, active infrared has also been used at the structural level to identify damage [206]. Infrared imagery has also proven useful for identifying areas on interest in powerlines [89, 207]. Infrared thermography has also been useful for building inspection using aerial vehicles as well as ground based techniques [208-210].

Though infrared has been the most widely used, in the future, different bands of light could add value to the SHM cycle.

2.6 Summary

The techniques presented in this chapter have been used in the research described in the following chapters and many research areas in the context of SHM. Image analysis techniques have been used for damage identification and measurement of measurements [99, 165, 173, 211-213]. Reconstructions from imagery have proven useful for many civil applications for mapping and analysis of damage in a global context [28, 37, 214]. The remote sensing and data analysis has the potential to revolutionize the field of SHM in the future [5]. Robotic platforms have the potential to aid in this goal and UAS in particular have the unique ability to access difficult to reach areas of the structure [215, 216]. Chapter 3 describes the objective of the research described in this thesis and the approach used to accomplish these goals.

Chapter 3 Objective, Hypothesis, and Approach

3.1 Research Objective

The objective of this thesis is to develop and validate techniques for acquiring useful information from unmanned aerial system (UAS) remote sensing platform to aid in the rapid and automated evaluation of infrastructure. The research described in this thesis is geared towards Structural Health Monitoring (SHM) applications with a particular emphasis on the types of materials and components found in transportation bridges and to some extent in buildings. This undertaking is crucial to determine how effective a UAS platform can be as a future inspection tool to provide decision makers with fast, reliable, and useful information to make more informed decisions about repair in a situation with limited resources.

The use of UAS for the purposes of bridge inspection is not new and there are several companies that are using UAS as data collection platforms [11, 30, 95, 217-219]. Furthermore, UAS have been used for the inspection of other types of infrastructure including pipelines [88], buildings [109, 152], power lines [89], and many others [220]. In addition, multispectral imagery has been used in many fields from the health of agriculture [221] to the mapping of terrain using satellites [191]. However, the biggest gap in SHM is obtaining information from the data acquired by any sensor [222, 223]. Hence, the research in this thesis focuses on the important step of acquiring data using UAS platforms, however it also emphasizes on the necessary processing of such data to help obtain useful information in the context of defects and damage commonly found in civil infrastructure.

The challenges associated with SHM in general are related to creating information from data across all of the monitoring and NDT techniques used. All of this information must result in an informed decision on how to maintain the structure. There is no single technique that can be used to determine what the structure needs to be properly maintained so that failure does not occur. Using UAS as an inspection tool will also not completely solve this problem, however, the data acquired from UAS can be further processed using developments found in related science and technology fields including remote sensing, digital signal processing, computer vision, and machine learning to provide more information more frequently which is a major benefit to the current state of practice. In many cases, inspectors take a diagram of the structure and draw on it to localize where the damage exists and mark in detail what they see. Though this is very reliable, it is also time consuming and very dependent on the experience and dedication of the inspection personnel. In addition, if a different inspection crew is used during the next inspection, it is often the case that the old information is not leveraged effectively. Intelligent acquisition of data and data processing have the potential, therefore, to revolutionize the way data is analyzed in an SHM setting with the use of robotic systems such as UAS. The Robotic Assisted Bridge Inspection Tool (RABIT^{TM}) and Targeted Hits to Measure Performance Response (THMPR), shown in Figure 37, are examples of how such robotic platforms are currently being utilized on structures to provide both data and information on the performance of the structure [20, 224]. Since UAS are noncontact sensor platforms, the challenges include collection of data, organization and visualization of data, missing data, and most importantly analysis of data. This thesis primarily focuses on the visualization and analysis of the data collected by UAS.

(a) (b)

Figure 37. (a) RABIT™ and (b) THMPR

3.2 Targeted Hypotheses

Three main hypotheses are explored in this thesis including that UAS can assist: (1) the identification of damaged areas on structures and present locally acquired data in a global perspective to automate detection and visualization, (2) the measurement of deformation in locations of interest, and (3) the demonstration that multispectral imagery can further assist in the quantification of deformation at different scales. The first hypothesis is explored with the intention to make the process of damage identification more automated and assist towards reliable and rapid damage localization, which is an important problem due to the complexity and length scales associated with infrastructure. The second hypothesis is interesting to explore as it aims to measure deformation solely using remote sensing platforms to drastically limit the need for installing sensors in difficult to reach areas of the structure, addressing a key factor that results in time and labor demanding SHM practices. The final hypothesis is made to extend the previous two by addressing the needed for reliable measurements at different distances to the target which is an advantage and currently a great challenge in the field of UAS due to their inherent navigation, control and sensing characteristics.

Both qualitative and quantitative measurements using computer vision, optical metrology, and computational modeling are needed to address these hypotheses, as outlined in Figure 38. Hence, the research challenges that are targeted in this dissertation include: (1) the identification of surface and subsurface defects and damage, (2) display local information on the global scale of the structure by stitching related information, (3) remotely measure deformation of structural elements using imagery, (4) isolate the movement of the UAS from the movement of the structure, (5) demonstrate that the technique will not hinder the current state of practice of visual inspection, (6) demonstrate that measurements can be obtained from multiple distances to the structure.

Figure 38 shows the general goals for the future of SHM leveraging UAS, starting with the communication and control of the system and ending with quantitative analysis of the data acquired leveraging computational models. Communication and control have been addressed in many senses with GPS and IMU systems, but more advanced control systems leveraging 3D SLAM or other high accuracy positioning systems without relying on GPS in the future will further extend the capabilities of UAS, particularly around civil structures were GPS can be unreliable. Remote sensing systems are very important to many applications within SHM, but are the most important sensors required for UAS. These systems acquire a lot of data quickly and though they are less accurate and often qualitative, they do not require installation when placed on a mobile platform and have the ability to point inspection personnel in the right direction for a more detailed analysis with a contact method. The big advantage of remote sensing in acquiring a lot of data is also a hindrance because it takes time to analyze the data. Therefore, signal processing and computer vision can be leveraged to automate this process. Optical metrology can further extend the

qualitative nature of images into measurements leveraging digital image correlation and photogrammetry. These measurements can then be used to update a computational model and aid in the decisions for maintenance and repair.

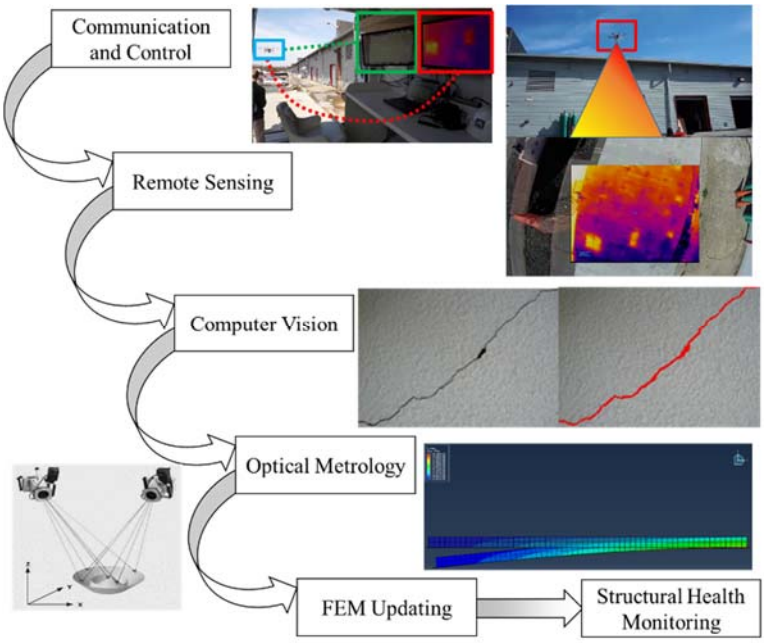


Figure 38. Future goals for UAS SHM

3.3 Challenges Addressed

3.3.1 Identification of Surface and Subsurface Damage

One of the biggest challenges that has kept image based remote sensing from widespread application in the field of SHM is the problem of extracting information from images. Images are generally only taken at locations were damage is identified during an inspection. This is important information, but different damage types have been commonly identified manually while they further require specific technical expertise to detect them. If a remote sensing platform is used to capture many images of the structure, damage would

also have to be manually identified in the images so no benefit is really gained. However, if the damage is identified automatically through the use of image processing and computer vision algorithms, the damage information can be immediately sent to the decision makers without the need to manually identify and document each area of damage. Furthermore, it could allow for direct comparisons to previous inspections.

The identification of surface and subsurface damage are described in Chapter 4 and Chapter 5. Chapter 4 introduces the developed post-processing algorithm used with UAS imagery to identify cracks leveraging the color and the gradients associated with the crack pixels. It also introduces the identification of corrosion using color and the k-means clustering algorithm. The results were compared to manual identification. Chapter 5 explains the identification of subsurface delaminations in a simulated bridge deck using UAS imagery. The algorithm takes another step to estimate the size of the delamination using the width of the deck as a reference.

3.3.2 Global Map of the Structure

Most images that are taken during an inspection are for documentation purposes only and do not give a good representation of the health of the structure because they are generally sparse due to length scale of the inspected structures. If the inspectors do not take careful notes on where the image was taken, then the collected images are useless regardless of whether or not they reveal damage. Chapter 4 and Chapter 5 explain the benefits to capturing images all across a bridge deck for the identification and localization of damage. The results of the test are the identification of damage and display on the global view of the structure such that the damage information is useful to the user. The challenges with creating the map can be mitigated with appropriate stitching software in some cases

such as Microsoft ICE [225], or in a more general case using VisualSFM [176]. In this thesis, an algorithm was developed using the output of VisualSFM to fit a plane to the structure and project the images taken in the field onto the plane. This development solves the problem of errors associated with the data capture of the UAS caused by UAS location errors. This method was further used to display damage information on the image mosaic.

3.3.3 Measurement of Deformation of Structural Elements using Imagery

Deformation of structural elements is an important metric for determining their overall health. It can be further used to determine if the structure is moving more than it was designed for, estimating the stiffness loss from a previous inspection, and thus it can help determine the cause of problems associated with the structure. Standard string potentiometers do an excellent job in determining the displacement of the structure at specific locations, however these systems require a ground reference which can be difficult to install. Additional issues with noise that appear due to environmental factors, such as wind, for instance when the cable connecting the sensor to the structure is long. Wireless and noncontact laser sensors have been applied, however they are generally expensive and have other problems associated with them such as range and power consumption. Chapter 6 explains how static and UAS imagery can be used to obtain remote deformation measurements of structures. Both fiducial point-based methods and full field techniques are used to determine deformation of beams in a laboratory environment. The point-based method can obtain point results directly comparable to displacement gauges. Full field techniques using digital image correlation (DIC) provide a range of values across the structural element which can be more useful for analysis.

3.3.4 Isolation of UAS Movement

In order to obtain measurements or relative measurements of a structure, a reference must be used to determine the exact position of the UAS. In Chapter 6, a static scale bar was placed in the scene which remained fixed throughout the test. This was required in all of the measurements obtained with a UAS. This is one major challenge to the application of this technology in the field, however it still has the ability to provide full field measurements which is an improvement over displacement gauges which also need a ground based reference.

3.3.5 Invisible Speckle Pattern

Another challenge associated with placing patterns or fiducial markers on a structure is the need to not hinder the process of current inspection methods. Applying a speckle pattern to a structure will make it difficult for inspection personnel to identify surface damage such as cracking. The simulated example shown in Figure 39 demonstrates this concept. The random pattern shown in (a) hides the crack as shown in (b) which can easily be identified in (c). In this context, work towards an "invisible" speckle pattern using multispectral imagery outside the visual range is explored in this thesis. This will be of vital importance to the implementation of such a technique to in service structures.

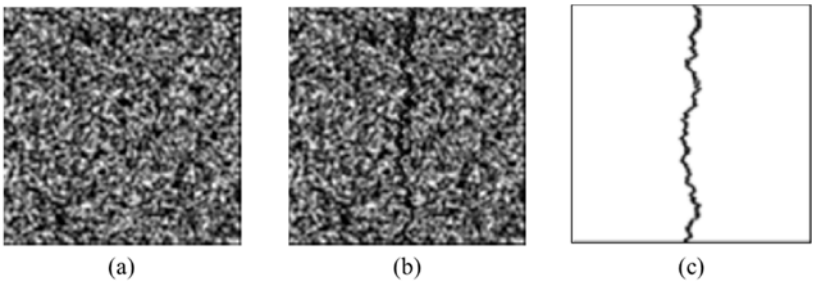


Figure 39. a. Random speckle pattern. b. Same speckle pattern with crack. c. Crack without speckle pattern.

3.3.6 Multiscale DIC Measurements

The need for an "invisible" pattern also leads to the conclusion that if a UAS is used to measure deformation using DIC, the optimal speckle pattern cannot be achieved when the UAS is flying at a different distance. This leads to the presentation of the multiscale multispectral DIC approach using different bands in the electromagnetic spectrum to produce multiple speckle patterns that are optimized for different distances from the target. This makes it possible for a UAS to fly further from the structure, determine an area of interest, and then fly closer to the structure to obtain a more fine measurement of the structure using a smaller speckle pattern optimized for the shorter working distance.

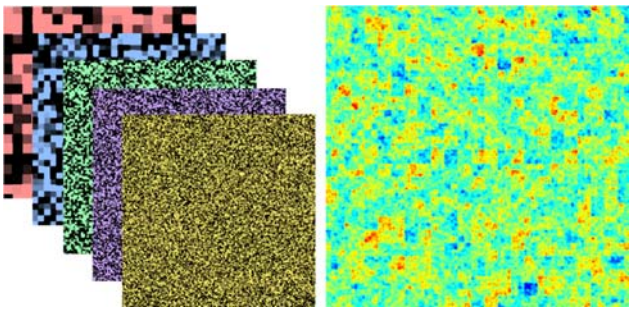


Figure 40. Multiscale speckle pattern optimized for specified distances

3.4 Technical Approach

3.4.1 Unmanned Aerial System

Multirotor systems were the types of UAS chosen for this research. They are easier to fly than many other systems, relatively stable with the use of flight controls, and they are capable of hovering in roughly the same location using position control. They can also carry multiple different remote sensing payloads. The exact UAS used in this dissertation are shown in Figure 41.

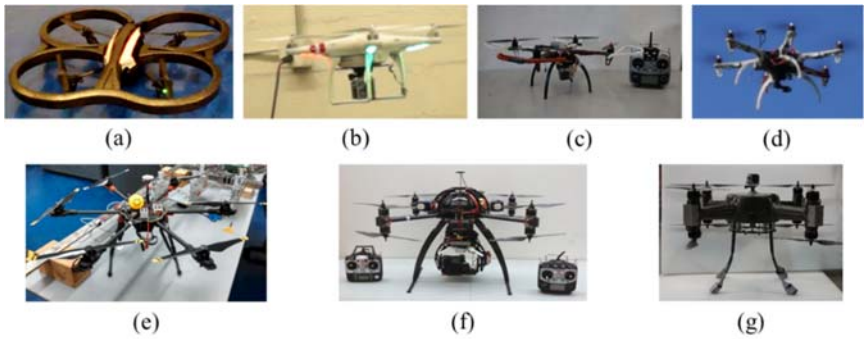


Figure 41. UAS used for this research (a) Commercial Parrot AR (b) DJI Phantom I (c) Constructed F450 (d) Commercial F550 (e) Commercial Tarrot 650 (f) Commercial Skyjib X4 (g) Constructed with Piasecki Aircraft Corporation.

For multirotors, the thrust to weight ratio should be between 2 and 3 in order to control the system since it uses the different motors to control pitch, roll, and yaw. When designing a system, it is important to determine the weight of its individual components and the amount of thrust that must be produced to lift the system and payload. In general, larger propellers produce higher amounts of lift. In order to spin those larger propellers, though, larger motors are also needed. All of the systems used for the research presented in this thesis used brushless motors which cannot directly run on the output from batteries. Hence, Electronic Speed Controllers (ESCs) were used as well, rated for the highest number of amps they can send to the motor. The batteries have a C rating which determines how much current they can discharge. All of these parameters had to be used to design the power systems of each UAS. The UAS shown in Figure 41 (a), (b), and (c) were limited to indoor use and relied on 3s batteries for power. The UAS in Figure 41 (d) used a 4s battery and Figure 41 (e), (f), and (g) used 6s batteries. The ESCs were all overdesigned to ensure the motors would not be limited during flight due to current throughput. Figure 42 shows the full electronic diagram of the UAS shown in Figure 41 (g).

In Figure 42, there are 8 motors and 8 ESCs, but for other multicopters, it is possible to use fewer. The colors of the dashed bocks correspond to the colors of the components in Figure 10 and Figure 11. This was the largest system designed and manufactured. The Pixhawk flight controller, GPS, and telemetry were used to control the UAS. It is the same flight control system used for the UAS in Figure 41 (c).

The systems used indoors did not leverage the GPS sensor. An optical flow camera was used for position control for the UAS in Figure 41 (a) and (c). The optical flow camera faced down for both systems and requires an ultrasonic sensor to estimate the height of the system. When used to hover, the sensors assume the height does not change and the motion of the system in horizontal and vertical directions determined from the camera are used to send a signal to the flight control to keep the system in the same position. In general, the ultrasonic sensors on these systems are only good for a few meters so optical flow breaks down at higher altitudes. Optical flow also requires that the ground below the system have significant texture to correctly estimate the movement.

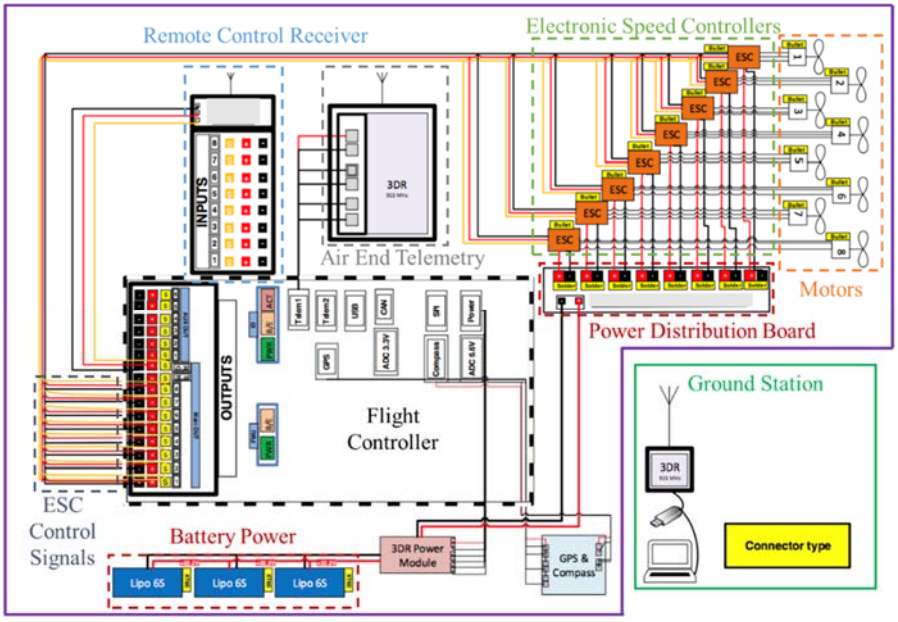


Figure 42. Electronic schematic of the UAS

3.4.2 Sensing Systems

There are many types of sensors used for SHM applications. Some contact sensors include strain gauges, accelerometers, and displacement gauges. Remote sensing systems including cameras, LiDAR, and wireless sensors. The systems used for this research were primarily different types of camera sensors and were validated with contact sensors as well as commercial imaging systems.

A Sony Nex 7, Sony a6000, Canon PowerShot Elph130 IS, and a GoPro Hero 3^{+} were the color cameras used for the experiments. Each of these systems were chosen for different experiments for different reasons. For instance, the GoPro camera is very lightweight, durable, and can be controlled from the ground so it was useful in experiments that required a UAS. The Sony cameras are of much higher quality, but they are much heavier and were therefore used as validation sensors. The Canon was a camera that had specifications which placed it in between the GoPro and the Sony. It was also more difficult to be integrated with a UAS and was heavier than the GoPro, but it was possible to control the camera from the ground using the remote control used to fly the UAS by mapping a switch to the channel in the flight controller that was attached to the camera. In addition, the Canon hack development kit (CHDK) was used to set up the image capture system [226].

A FLIR 325sc, FLIR Tau2, and ICI9320 were the infrared cameras used for the experiments. The FLIR 325sc has better sensitivity and outputs radiometric temperature measurements, however it is also very heavy. It was used mainly as the validation camera for the FLIR Tau2 and ICI9320. The FLIR Tau2 does not output temperature measurements, but captures video at 30fps and is lightweight. It was used in several test

flights on a UAS. The ICI9320 camera was also used on a UAS and does output temperature measurements. The camera was used with a UAV module which allowed for the remote capture of images at a maximum of 1 image per second.

An X-Box Kinect was also used during some of the experiments to simulate a LiDAR system. It is significantly less expensive than other LiDAR systems and provides similar information. Another advantage of the X-Box Kinect is that the sensor has a color camera and a distance map to each pixel in the image. This allows for 3D measurements to be obtained leveraging the two sensors.

3.4.3 Applications

The applications of the technology explained in the previous sections are focused on structural assessment in both lab and field settings targeting bridges, however the methods and applications developed can be extended to buildings, as well as other infrastructure examples including powerlines, telecommunication antennas, wind turbines, pipelines, storage tanks, etc. The systems were applied to indoor structures including building walls, cantilever beams and a model of a metallic bridge deck grid to control some of the parameters involved in such measurements. The experiments performed in the lab with UAS were limited to the smaller UAS shown in Figure 41 (a), (b), and (c) due to space restrictions. The outdoor experiments using the UAS were performed on a simulated bridge deck with manufactured defects that was used to test different methods used for NDT. The experiments what were performed in the field did not use UAS for legal and safety reasons.

3.4.4 Software

There were several sets of software used to perform the experiments presented herein. MATLAB was the primary software used for the image analysis algorithms.

ABAQUS was used to perform all of the finite element analysis. LabVIEW was used for data acquisition for the traditional sensors used for comparison. Furthermore, the commercial optical metrology systems used for the experiments were ARAMIS for full field DIC and TRITOP for photogrammetry, both developed by GOM.

3.5 Thesis Results Overview

The results presented in this thesis are broken into three parts: (1) identification of damage, (2) deformation measurement, and (3) multispectral approach to deformation measurements.

3.5.1 Identification of Damage

The identification of damage relies on image processing for structural element identification and localization, identification of damage such as cracks, and quantification in the form of a size estimation. The combination of different types of imagery is expected to provide a more complete idea of the health of the structure. In the case of bridge deck delaminations, it is important to distinguish between surface artifacts and subsurface defects. A combination of the color and IR images will be used to identify the surface damage and use them to eliminate false positives in the IR imagery. The plan for damage identification is displayed in Figure 43.

Surface damage identification for crack identification and the processing associated with creating an image mosaic to create the context for the damage is presented in Chapter 4. Chapter 5 demonstrates the use of infrared imagery with both radiometric and nonradiometric data for automated identification of delaminated areas in a bridge deck. These techniques were also demonstrated in a field setting on a ground platform to prove the concept.

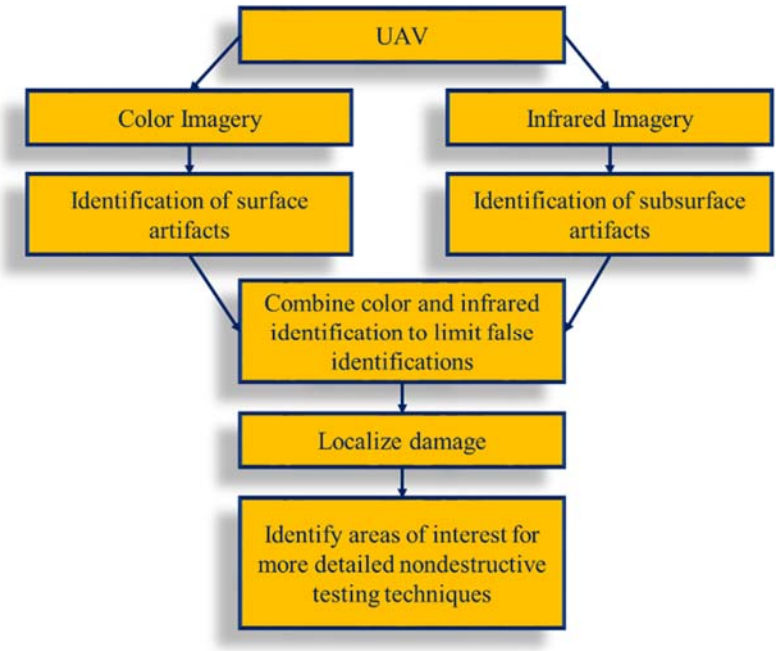


Figure 43. Framework for damage identification

3.5.2 Deformation Measurements

The measurement of deformation using target based photogrammetry first identifies the targets on the structure in multiple images, calculates the 3D coordinates of each point, then the structure was loaded and a second measurement was collected and compared to the first. The points of interest were fixed markers used as references for deformation and the remaining markers were used as deformation measurement points. The measurement points were either moved by a predetermined amount or had a corresponding, high accuracy, measurement systems associated with each marker which were used to assess the results of the methods. Figure 44 shows the plan of action for the measurement task.

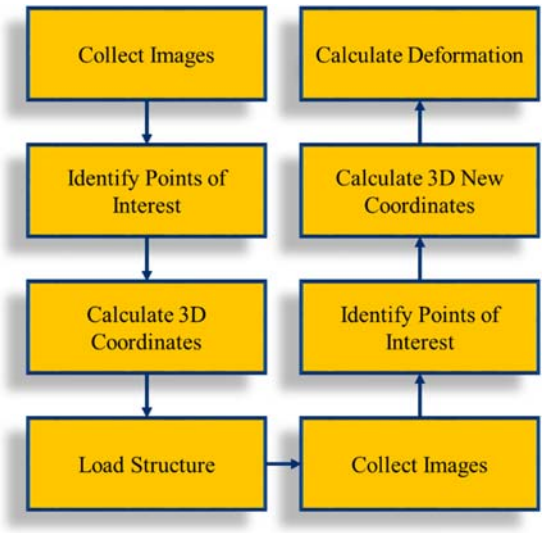


Figure 44. Framework for measurement and update

3.5.3 Multispectral Approach for Multiscale and Selective Imaging

A major goal in this task is to identify paints and tailor multiple speckle patterns for a specified working distance to perform deformation measurements. An example of a multiscale pattern is shown in Figure 45 where the red dots are visible at the closer distance shown in (a) and the red dots are not visible from the farther working distance because of aliasing shown in (b) [227].

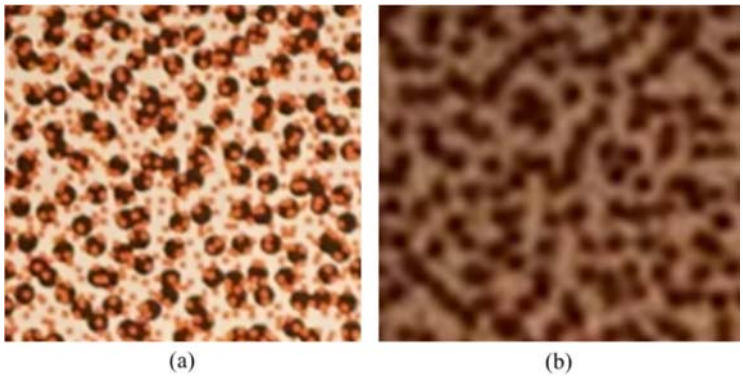


Figure 45. Multiscale speckle pattern (a) close working distance and (b) far working distance [227].

Multiscale speckle patterns can be accomplished in multiple ways, but the approach chosen was to paint a fluorescent paint on top of the initial speckle pattern in order to help distinguish the two patterns more easily. This could also be accomplished by tailoring a material to only appear in predetermined wavelengths of light, but would also require the analysis of the resolution of the camera based on the wavelength chosen. Figure 46 shows the framework for collecting and analyzing the multispectral imagery.

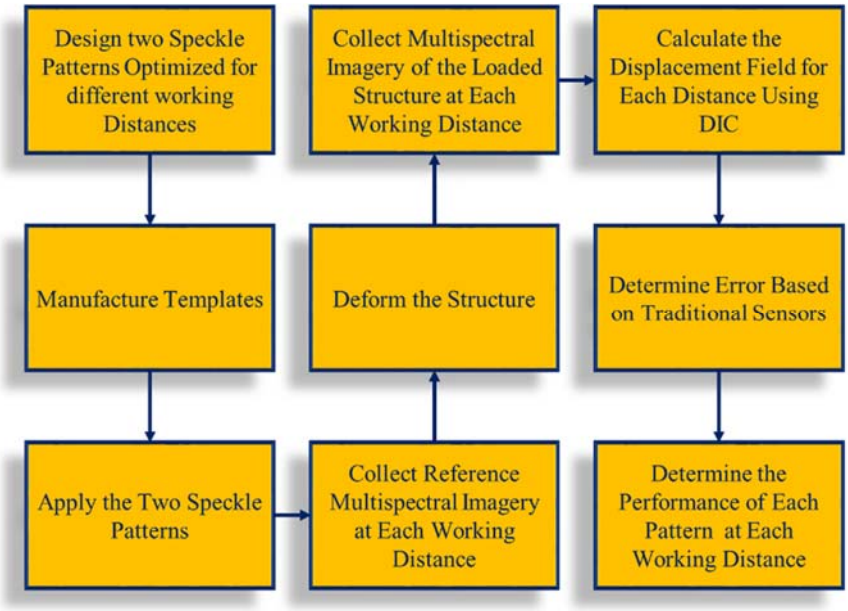


Figure 46. Framework for multispectral imaging

Chapter 4 Surface Damage Identification

4.1 Introduction

Aerial image collection has been demonstrated for both military and civil applications, but commonly provides only qualitative assessment completed by users, often with manual identification. The representative tests herein are designed to demonstrate that both automated qualitative identification and quantitative measurements can be attained through UAS imagery. Specifically, identification and quantification of corrosion areas and their size, as well as localization of cracks leveraging images collected by ground and UAS based platforms are presented in this chapter, since these surface defects are commonly found in infrastructure. The obtained results demonstrate the usefulness of UAS acquired imagery for SHM applications. Furthermore, visualization of the damage information obtained from an in service structure is also presented in an image mosaic manner, to assist with the task of data handling and their presentation.

Surface damage can take many forms, shapes, and sizes and can arise from different sources. In addition, many issues may affect aerial imaging such as shadows, paint, and debris. These challenges make automated identification of damage in imagery an open area of research [40]. Despite the challenges, aerial imaging is a technique that can be applied to many types of infrastructure such as buildings, roadways, pipelines, and bridges [105, 220, 228]. Visual inspection of masonry buildings, for example, has been conducted with a focus on obtaining clear images and video while flying [92]. In terms of on-going research, the flight path of a UAS has been investigated to determine ways to minimize the effect of varying light conditions and blurring caused by wind moving the system during

image capture [30]. In addition, the application of three dimensional (3D) sensors on a UAS platform have been proposed with the objective to create 3D maps of building facades [91]. Furthermore, utilizing image processing and identification, 3D models have been extracted from imagery at both small and large scale [28, 29]. Specifically, the identification of the global location of damaged areas is crucial to e.g. inspection personnel and decision makers. In addition, damage location often must be properly documented by inspection personnel in order to make use of the information. The same concept applies to images obtained by UAS [229]. If the image provides no context as to where it was taken from or to what exact location on the structure and within the field of view it corresponds to, then it is practically useless for inspection purposes. Furthermore, images and video on a structure obtained by UAS can be disorienting when being viewed after the flight and therefore, an automated method for easier visualization is necessary to make use of such data [215]. In this direction, several image panorama, mosaic, and stitching algorithms have been developed using different assumptions and types of camera movement [149]. In such approaches, the presentation of the data acquired is the most important part to convert it into useful information to decision makers which is a major goal of SHM [223].

Cracks constitute a common form of damage in many structures and for this reason, several methods for image based crack detection have been researched [230]. Examples of semi-automated approaches include the fly fisher and route finder algorithms which require user input for each crack to find the path of the crack given the two endpoints [231]. The percolation approach has also been used successfully to identify cracks using a region growing algorithm to eliminate background pixels by assuming all foreground darker pixels are cracks [130]. Furthermore, periodic image noise removal has been used for crack

identification using Fourier transform [232]. Moreover, edge detection approaches have been implemented for many purposes including crack identification since they are capable of highlighting areas of high gradients within the image [233, 234]. In this context, the Prewitt edge detection method was modified to identify cracks by attempting to fit lines on the crack path [235]. A multi-resolution Laplacian edge detection method was explored by Pei et al [236]. In addition, Zhang and Li et al. used a matched filtering algorithm developed by Chaudhuri et al [237] for blood vessel identification in retinal images which uses predefined filters at varying orientations to identify pavement cracks [238]. Tensor voting has also been applied to identify crack connectivity while shadow correction was additionally used to improve the results of this method [239, 240]. Besides cracks, image processing techniques have also been applied for corrosion monitoring [151, 241]. Identification using color was suggested to determine if blasting was required [242].

Object identification using UAS imagery for infrastructure purposes is also an active area of research [216, 243]. In this area, a morphological and thresholding methods have been used to identify cracks on a masonry building from UAS imagery [87]. Active contours, edge detection and morphological operators were also used to identify cracks in a competitive study with UAS imagery [244]. For example, edge detection was used with UAS to identify cracks in buildings and bridges [11, 109, 152], as well as for crack identification on steel structural elements [245]. Reagan et al. demonstrated that full field displacement measurements obtained with stereo cameras on a UAS could be leveraged to obtain crack opening and closing [246, 247]. UAS have been used to identify problem spots in unpaved roads [248, 249]. The use and effectiveness in the identification process of these algorithms depends heavily on the objects being identified, on the dataset in which the

algorithm applied, and the environmental conditions during the experiments, but each of the developed methods have shown great promise in identifying areas of interest for further analysis or repair. Furthermore, algorithms developed for UAS applications can be used on imagery obtained by other robotic platforms or by inspection personnel.

For the experiments explained in this chapter, images were obtained statically on the ground as well as in the lab with a UAS. The UAS images were saved onboard and post-processed for the experiments described. The experiments that leveraged UAS in this chapter were performed indoors where GPS signal was not available to provide position information. It should be noted that the analysis techniques presented do not depend on the sensor position. However, if the reported measurements were performed outdoors using a UAS equipped with high accuracy GPS and appropriate gimbal units, it would be beneficial to have the additional, high fidelity, position information at the time of image capture. This would allow for additional information including time stamps, position coordinates, and the specific angles at which the images were taken, which could all be leveraged in the data processing and future inspections.

4.2 Preprocessing

As with all cameras, lens distortion correction was required for all images used in this chapter. It should be noted that the GoPro lens causes significant lens distortion due to its 170º ultra-wide angle lens with a focal length of 2.3mm. The GoPro camera had a lens and camera settings that could not be changed. The camera calibration algorithm described in section 2.4.3 was leveraged, using 50 images of a checkerboard with the camera locations displayed in Figure 47. The algorithm allowed the user to analyze the error of the model for each image using the known locations of the corners of the checkerboard. Three

standard deviations were used to analyze the error and each of the error parameters was reduced to less than one pixel after the error analysis, which resulted in sufficient quality to measure objects in the corrected images. Ordinarily, cameras with a smaller field of view and a longer focal length will have errors less than one pixel on the first or second attempt because the distortion is less severe than with wide angle lenses. In this case, the error analysis required several attempts to reduce the error associated with the parameters.

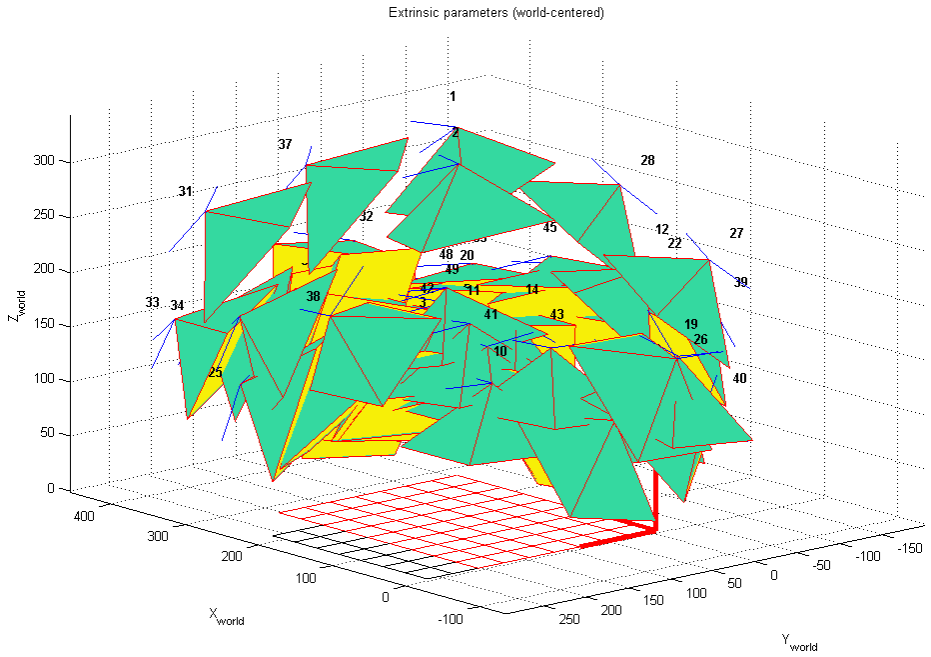
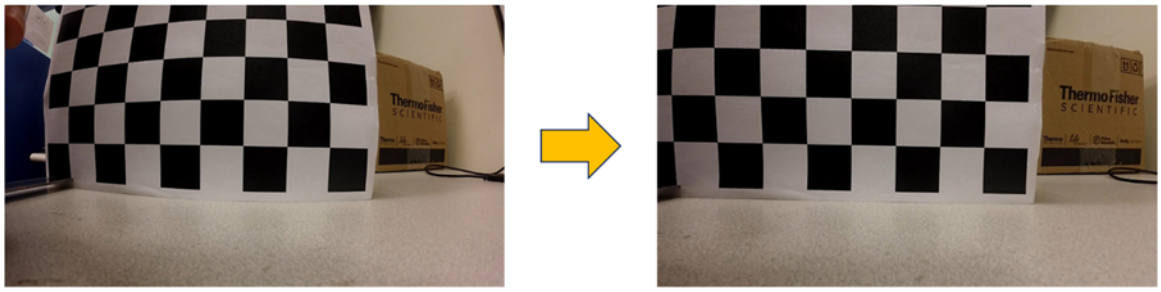


Figure 47. Camera calibration result for the GoPro camera [159]

After the equation was determined from the algorithm, it could then be applied to new images captured with the same camera and lens system to correct for the lens distortion. For example, the improvement in Figure 48 (b) is clear especially around the edges of the image.



(a) (b)

Figure 48. Results of GoPro lens distortion correction

The lens distortion corrected image appears to have less field of view than the original image. This is due to the edges of the images obtained by this camera were cropped to remove black portions that were caused by sections of the original image that were not mapped into the field of view of the corrected image. The Parrot AR 2.0 camera parameters also could not be changed and consequently the lens distortion associated with the camera also had a large amount of distortion. Hence, the same method was used to also correct these images. The next step in preprocessing of the images was to apply the homography matrix, explained in section 2.4.4, to project the image onto a plane within the image, which was required to obtain the correct size of the identified damage.

4.3 Corrosion

Corrosion is a type of damage that affects many types of infrastructure. This type of damage has been identified using images combined with machine learning based on a robotic platform [242]. This type of damage is particularly difficult both in terms of identification and measurement based on imagery because it does not have a predefined shape or size associated with it and can often blend into the background of the images. Though color is an easy way to detect this type of damage, it is a difficult metric to apply as a threshold because different lighting conditions change the range of colors the camera

captures. Therefore, a threshold that appears to work well in one image or image set may not work well in another of the same scene.

In order to simulate the corrosion in the lab, an image of an actual corrosion was found online [250], printed, cut out to fit the size a steel grid, and taped on the grid to simulate a corrosion location on the steel beam that changes over time. The UAS used for this experiment was a DJI Phantom I in stabilize mode with a 10MP GoPro Hero 3^{+} Silver edition fixed to the bottom of the system. The images were acquired manually with the cellular phone application. The pilot controlled the UAS to ensure that it was hovering next to the grid while taking images of each stage. After landing the system, the simulated corrosion was removed and resized two more times to become smaller and the process was repeated. Two representative images obtained from the UAS during the flight are shown in Figure 49.

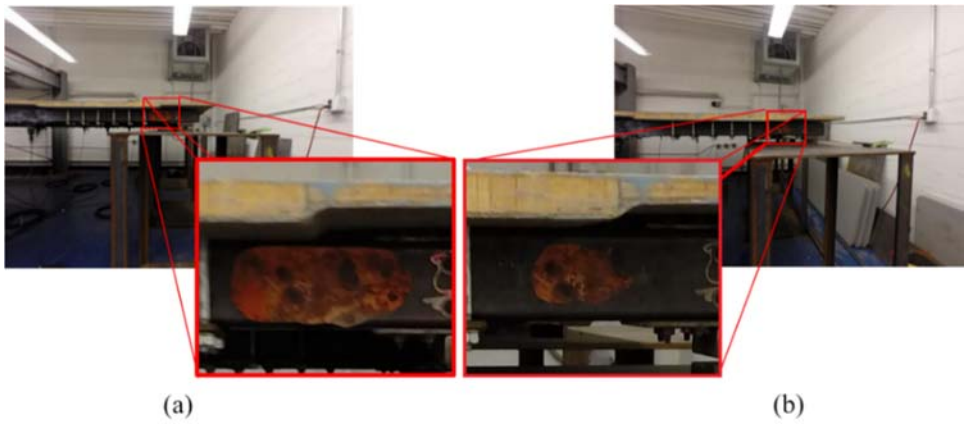


Figure 49. Corrosion setup on steel grid. (a) Image of the larger corrosion. (b) Image of the smaller corrosion [251].

The purpose of this test was to identify and obtain size estimates of the corrosion. The images of the corrosion shown in Figure 49 were enlarged and highlighted in the red

boxes to show the area of interest during the experiment and highlight that the corrosion size was changed. Both manual and automated methods were used to determine the size of the corrosion in the UAS images. The manual measurements were completed by counting pixels in the image that appeared to be the corrosion and the automated method used the K-means algorithm to select the areas of corrosion. The images were corrected for lens distortion while the bearing plates were used as the reference measurement and for the homography matrix calculation. A ruler was used to determine the actual corrosion size, which is actually the way that such measurement would be performed during an inspection. Measurements using the UAS imagery were made by placing the simulated corrosion on a high contrast background and taking a high resolution image perpendicular to the surface with a ruler in the image. An example of the ground truth images are shown in Figure 50 where Figure 50 (a) shows the image and Figure 50 (b) shows the mask used to determine the area of the corrosion.

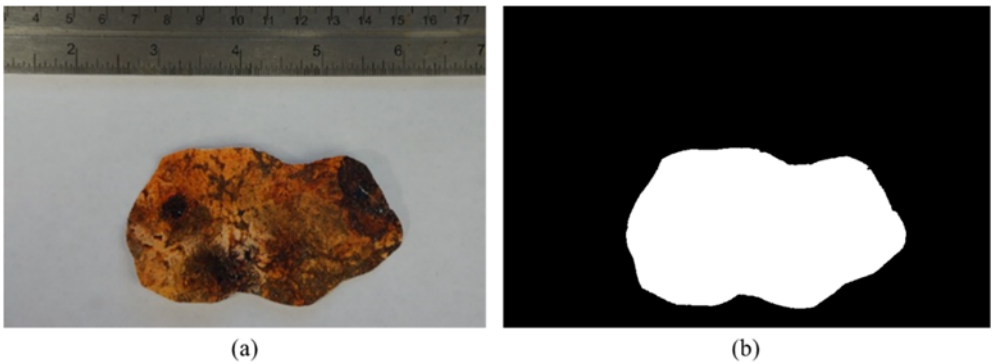


Figure 50. (a) Example of image used for the exact measurement and (b) the processed image used for size estimation

Table 3 shows the results of the corrosion experiment where the small corrosion was assumed to be the initial state for all of the measurement types. The UAS manual, UAS

k-means, and the static measurements measured the area in pixels and converted the pixel measurements to millimeters using the distance per pixel measurement in the plane. These measurements are dependent on the user inputs to form the homography matrix. The static measurements are assumed to be the most accurate due to the high resolution static images, the short working distance, the high contrast between the corrosion and the paper, and proximity of the scale bar to the corrosion. It is for these reasons that this measurement was chosen as the ground truth rather than the measurements taken with the ruler.

Table 3. Results of Corrosion Test

Measurement	Ruler			UAS Manual			UAS K-means			Actual	
Set Size	Area (mm)	Growth	Error	Area (mm)	Growth	Error	Area (mm)	Growth	Error	Area (mm)	Growth
Small	3,811	0%	15%	3,632	0%	10%	3,739.3	0%	13%	3,308	0%
Medium	6,452	169%	-1%	6,342	175%	-3%	5,784.5	155%	-12%	6,550	198%
Large	10,161	267%	5%	9,312	256%	-4%	10,247.5	274%	6%	9,699	293%

The measurements obtained manually from the UAS images were found to be more accurate than the K-means. However, both measurements were within 15% of the actual size. The K-means method suffers from repeatability issues and does not always achieve the correct measurement, however, it is also far less labor intensive which potentially makes it a more suitable approach for field measurements.

4.4 Crack Identification

Another major form of damage that is critical to monitor with SHM techniques is cracking [12, 13, 18, 96]. Similar to challenges with corrosion identification in images exist with crack identification. However, one property of cracks that allows them to be identified

easier is the width of the crack is in almost all cases small, but this also is a challenge because the resolving the crack is dependent on the working distance of the camera and the size of the crack [174]. Two experiments were performed to determine how cracks could be detected using a color camera from multiple distances to simulate a scenario where the UAS camera could resolve and detect a crack on civil infrastructure. The first experiment focused on quantifying the effect of camera distance on the viewing platform of the pilot using ideal, simulated cracks of varying size, as shown in Figure 51 (a) using a 16 megapixel camera using both the screen for the live view and post processed with a computer. A sheet of paper with lines of different controlled thicknesses was used to simulate cracks and the camera moved to different controlled distances from the target. The second experiment used a Parrot AR 2.0 UAS camera to monitor the cracks developing on a masonry wall during cyclic loading [252]. Due to the space limitations, the UAS could not be flown safely and therefore images were taken statically for the measurements. Table 4 shows the results obtained for both "live" and "post-processed," since digital zoom was not available to the pilot in real time using this camera and UAS setup.

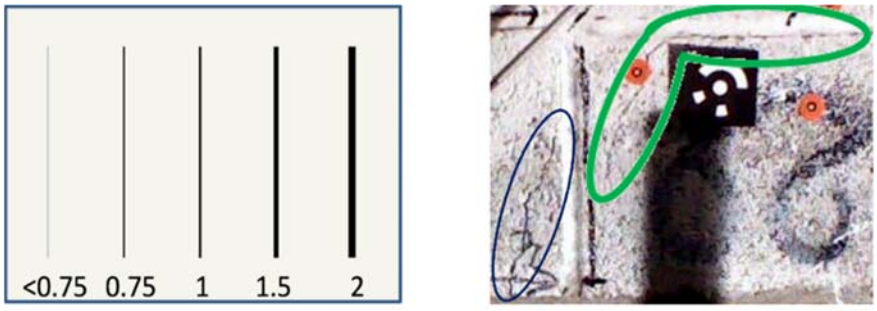


Figure 51. (a) Simulated Cracks on a sheet of paper for visual identification using a digital camera and the camera of the UAV; (b) Examples of cracks on an actual wall made of partially grouted concrete masonry blocks.

The results shown in Table 4 indicate that the smallest line on the sheet with thickness <0.75 mm could be observed at a working distance of 12 meters after post-processing. Furthermore, images without post-processing showed that at 9 meters, the smallest observable crack was 2mm wide, while the smallest crack with width <0.75mm could be detected in the "live" mode at a distance of about 3 meters from the target. This demonstrates that simply using the pilot or observer to identify the cracks real time on a screen is not as effective as the image processing after the flight.

Table 4. Crack detection results of distance test shown in Figure 51 (a)

Crack Detection Analysis	Distance (m)	3.0	4.6	6.1	7.6	9.1	10.7	12.2
Live	Smallest Crack Thickness (mm)	<.75	0.75	0.75	1.5	2	N/A	N/A
Post-Processed		0.75	0.75	0.75	0.75	0.75	0.75	0.75

Figure 51 (b) shows a representative image of crack detection on a masonry wall using the built in Parrot AR 2.0 camera. The wall had dimensions of 4.2m x3.7m 0.2m and was loaded horizontally at the top of the wall by imposing cyclic displacement of increasing amplitude to simulate earthquake loads [18]. The highlighted areas in the image are the locations of cracks initiation on the partially grouted concrete masonry wall. It was determined that the UAS camera had a high enough resolution to allow the viewer to clearly see the cracks on an I-Pod Touch screen through the Parrot AR 2.0 application. The image quality is expected to have streaming resolution similar to what a pilot would have in the field using commercial equipment since lower resolution is required to achieve video streamed real time to the pilot with minimal lag time within the video. As the distance increases, the pilot would be expected to have a more difficult time identifying the cracks

based on the experiment from Figure 51 (a). At greater distances from the target, post-processing is necessary to identify the cracks until they are aliased out completely and therefore undetectable.

The results of these experiments were qualitative, with a goal of determining if a pilot could identify a crack in an image streamed to the ground real time alter the flight path to focus on that area of the structure. In many UAS setups, the pilot is responsible for triggering the camera and flying the system. As a result, if the pilot is unaware of the crack, it is unlikely that an image will be obtained from the UAS during the inspection and therefore not documented as a potential area of interest. This problem could potentially be mitigated if the entire structure is scanned. However, the resulting amount of data is much larger and a significant amount of data is not needed. In this context, an automated image analysis is required to help reduce the amount of data a user must search through.

A lab experiment was performed using a DJI Phantom I and a GoPro Hero3^{+} Silver Edition to capture images of cracks during flight. Indoor masonry cracks were used for this test and post-processing image techniques were developed to identify the cracks from the UAS imagery after the flights. Many crack detection algorithms are tested on images where the surface being analyzed is the only object in the field of view. These algorithms often rely on the color of the image or the edges of the image which can be unreliable when other objects are in the field of view, particularly in the complex environment expected in the field. Most UAS imagery contains areas within the field of view that are of no interest for crack identification [100]. It is for this reason that an algorithm was developed to mitigate some of these challenges. The crack detection algorithm was leveraged on images collected

by a UAS in the lab. The algorithm pseudocode is displayed below. It takes into account both the intensity of the pixels and the gradient to fill in the cracks that are identified.

```
Load image
Segment image
Convert to grayscale and apply median filter
for all rows and columns of the image
        Find the average and standard deviation of a specified neighborhood of pixels
        throughout the image
        Set cutoff criterion based on average and standard deviation
        if any of the current four pixels < cutoff
                Calculate and store the eigenvalues
        end
end
Remove edges created by segmentation from the eigenvalue map
Connect potential cracks with image dilation and erosion
Filter with shape criteria
Highlight cracks in the image using color to the crack pixels
```

After loading the image into MATLAB, it was segmented by color using the K-means algorithm to eliminate parts of the image that do not contain cracks. Since the concrete wall in question did not have color, only pixels with colors similar to grayscale were kept for analysis. Shadows caused a gradient of color that can cause false positives in many algorithms, so a median filter was applied to the images and the outcome is subtracted from the original image which reduces the effect of shadows in the image [232]. The Eigenvalues were then extracted to identify pixels with strong directional components in one direction. Both local and global criteria leveraging the mean and standard deviation of the eigenvalues were used to limit the effect of surface texture on the output. Furthermore, multi-resolution edge detection leveraging a median filtering was applied to remove edges that had interfaces with different materials caused by other objects in the scene [236]. Cracks larger than the filter size would then be eliminated, so careful consideration was

required before choosing a filter based on the image resolution and the camera working distance. Image morphology, described in section 2.3.2, was then used to connect the segments that were likely to be cracks [118]. Then area and perimeter were used to filter the image with shape criteria to remove segments that did not belong to cracks and finally, the cracks were highlighted in the original image. The roundness criteria is shown in equation (2.11) was used to eliminate an blob that had a roundness value of greater than 0.6 assuming that it was not a crack [253].

The crack identification algorithm presented above was validated using several images obtained from the internet and analyzed before testing it with images taken with a UAS. Twenty images with and without cracks were analyzed with different levels of complexity in each image. The results of the images with cracks are shown in Figure 52 and without cracks in Figure 53.

The crack images shown in Figure 52 (a), (b), and (c) were identified properly as shown in (e), (f), and (g) with little to no noise in the background. The crack shown in Figure 52 (d) had a missed identification at the very bottom of the image as shown in (h), but still successfully identified the majority of the crack. This was one of the most difficult images in which the algorithm was tested.

Figure 53 demonstrates how the algorithm performed with a lenient shape filter, but if a more stringent shape filter was applied, most of the noise would disappear. Ideally, no pixels in these images would have been identified as a crack. However, a more stringent shape filter could also cause problems with identification because missed identifications would increase in the crack images.

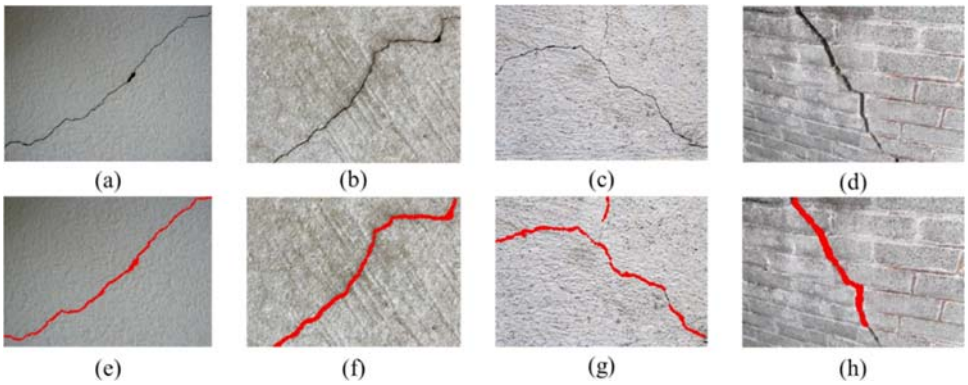


Figure 52. Original images (a-d) [254-257] and resulting images (e-h) after applying the identification algorithm

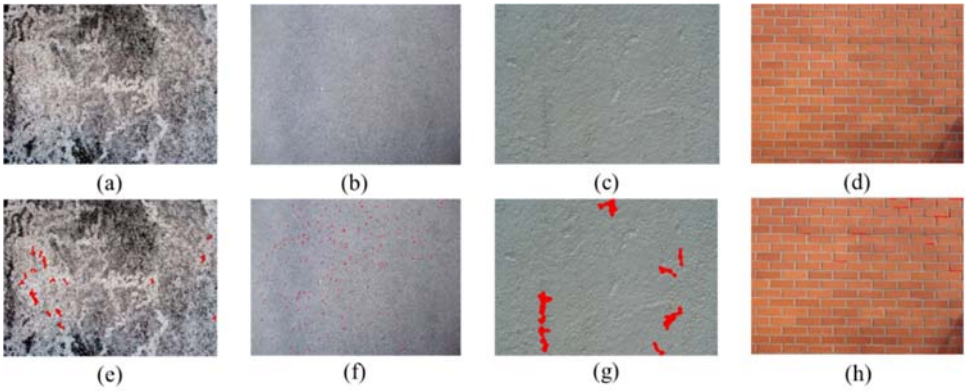


Figure 53. Original images (a-d) [258-261] and resulting images (e-h) after applying the identification algorithm

After the validation of the algorithm, the UAS was flown in the lab to collect images of wall cracks. Figure 54 (a) shows an image of the UAS hovering while the GoPro was capturing the image shown in Figure 54 (b). The camera was triggered remotely using the GoPro app for remote image capture. The image was corrected using the camera calibration algorithm as shown in Figure 54 (c) and the cracks were manually identified as shown in the red box. The results of the identification algorithm are shown in Figure 54 (d).

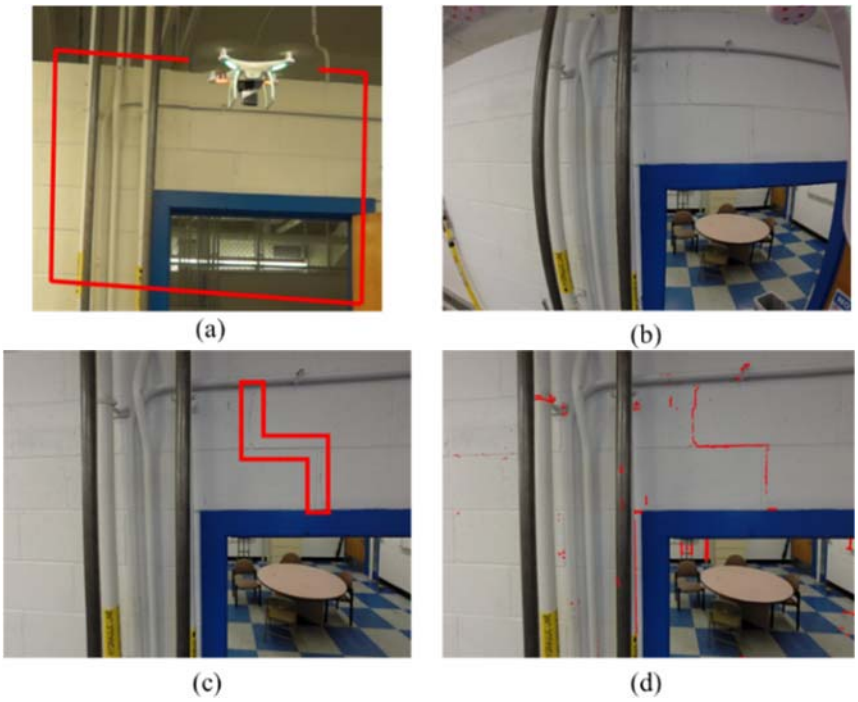


Figure 54. (a) Image of the UAS hovering to obtain image in (b). (c) Image in (b) corrected for lens distortion with manually identified cracks. (d) Results obtained from the crack detection algorithm.

The next major step after crack identification is measurement using known features in the image. Quantitative measurements were completed using the concrete blocks in the image as a scale and the physical size of the pixels in the image shown in Figure 54 was determined to be $0.3mm^2$. The cracks were also manually measured with a crack card and the crack sizes were almost all sub-millimeter with some crack widths less than the size of a pixel. A narrower field of view with a dimensional reference in the image would be required obtain accurate crack measurements. In Figure 54(d), the crack was correctly identified; however, there were several other features in the image were also falsely identified as cracks due to the image complexity compared to sample images obtained from the internet. Though the images are not as complex as images taken in the field, they are more representative of some of the challenging conditions that are encountered in the field.

4.5 Global View of a Structure

In the case of large structures, it is important to know what is in the field of view of each image that is acquired. However, the challenge in many field applications of imagery is the lack of record or knowledge of where the image was taken and the purpose of taking the image. In many cases, high quality imagery loses its value because they have no context to determine what parts of the structure appear in the field of view. One simple way to obtain a good view of the structure with the context for each image is image mosaicking [149].

The bridge deck is responsible for providing a driving surface for vehicles, distributing traffic loads to the other structural members, and protecting the structure. Figure 55 highlights the different parts of the bridge structure and highlights the location of the bridge deck. It is clear from the image that the bridge deck is the part of the bridge that is most exposed to the elements and traffic loads. As a result, the bridge deck accounts for 50% up to 80% of the maintenance funds [20]. There are several methods that are used to assess the health of bridge decks including impact echo, ground penetrating radar, infrared thermography, and chain drag. These methods often benefit from applying a grid pattern on the structure to aid in the localization of the damage to make documentation easier.

Figure 55. Image of a bridge structure with the different parts labeled.

If photos are captured of the entire bridge deck, this can serve as a schematic for damage documentation rather than the use of a sheet of paper that is used to document surface and subsurface damage. The field study was focused on NDE methods including impact echo, ground penetrating radar (GPR), infrared thermography (IRT), impact testing, and modal analysis. The grid was required for these methods, but they also served as the control points for the image mosaic. The painted points were spaced every two feet throughout the structure. These points were used to manually stitch the image together to obtain highly accurate representation of the deck. An example of an image mosaic using 383 images of an in service structure during a field study is shown in Figure 56. The images were preprocessed using lens distortion compensation described in section 2.4.3. Then, using the grid points painted on the structure, homography, described in section 2.4.4, was used to project the images onto the plane of the deck. The surface damage that was manually documented using a diagram of the deck was necessary during the test, however after the final image mosaic was produced, it was clear that this was a much better option because the damage size, shape, and locations are easily visualized and the locations of prior patching are also easily identified. Furthermore, it limits the possibility of mistakes in position of the damage while entering the data onto a map of the structure. This documentation is critical during the inspection and if any damage area is missed on the diagram, the images can then be analyzed after post processing of the NDE data to determine if there was in fact some surface anomaly that was influencing the other NDE data.

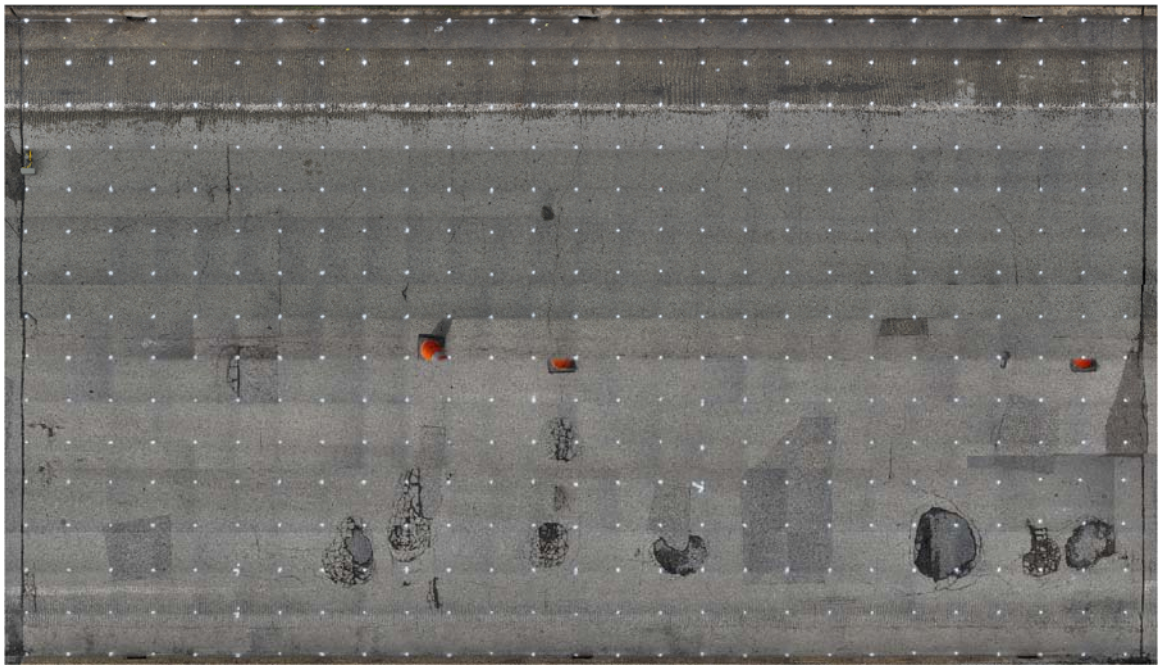


Figure 56. Field validation of image mosaic

To further simplify analysis, other NDE data could be overlaid on the structure to allow for the positioning of the surface damage with respect to other methods like GPR, IRT, or other NDE methods to gain a better understanding of what cause anomalies in the NDE data. After the field experiment, a graphical user interface (GUI) was created to help select areas of interest in the stitched image and pull up the original image in full resolution as shown in Figure 57. This would aid in the visualization of surface damage providing both high resolution imagery using the original images and the global context of the image. In the future, other NDE data could also be incorporated into the GUI for better localization and quantification of potentially damaged regions.

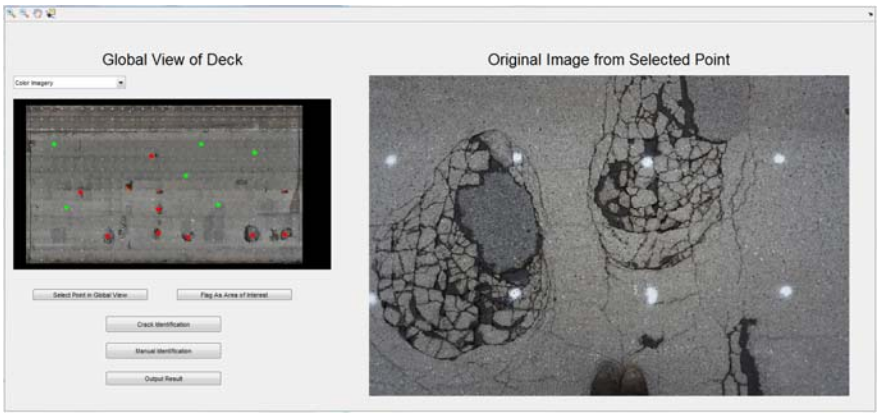


Figure 57. GUI for global damage identification

The GUI allows the user to click points in the stitched image. The points chosen are highlighted in green and if there is damage, the user can flag the area of interest in red. This type of interface is beneficial to a user because it allows an experienced inspector to analyze the structure without being on the structure which would save time, money, and potentially make the results of visual inspection more consistent.

The images in this image set was by far the most challenging for damage identification due to the large amount of exposed aggregate in the deck. The background color variation was removed using fast Fourier transform (FFT). The low frequency content was isolated from the image and removed from the original image because cracks mostly appear in the high frequency content due to the much higher gradients associated with them. A local peaks over threshold (POT) method was used to process the images following the POT method described in section 2.3.4 after adjusting the threshold values and an inverting the image [134]. A local method was chosen to ensure that the aggregate would not cause as many false identifications as the global threshold based method. The image was inverted because cracks are dark in comparison to the background. A shape filter and morphological operations were used to reduce the number of false positives as well.

```
Load image
Convert to grayscale
Invert the image
Correct with FFT by removing low frequency content
Set image block size
For each block in the image
        Compute peaks over threshold method
        Find all values greater than the threshold
        Perform shape filter
End
Perform morphological operations
Perform shape filter
Identify cracks
Save image
```

A few examples of the results of the crack identification algorithm are shown in Figure 58, Figure 59, and Figure 60. The top images in each figure are the original images and the bottom images are the images with the cracks identified. Figure 58 shows images that have fewer difficulties due to the conditions in the field. There are really only cracks in the images which makes them easier to identify.

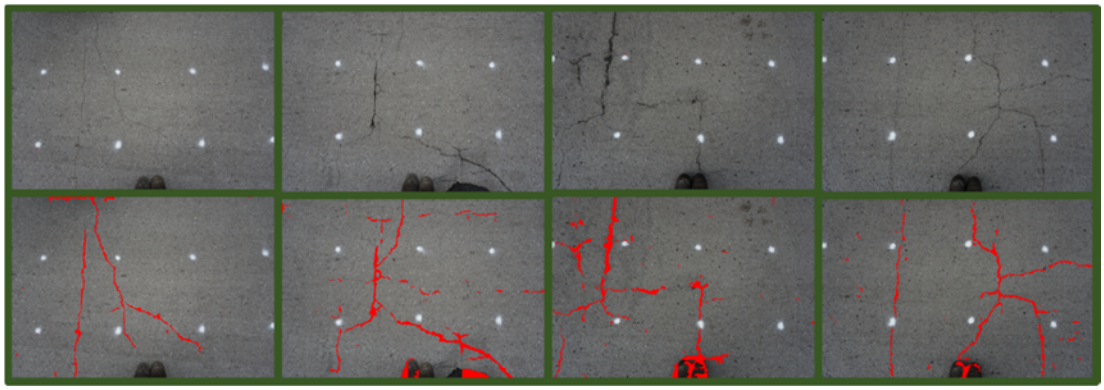


Figure 58. Sample of crack identification with good results. Top: Original image. Bottom: identified cracks

Figure 59 shows images with more difficulties including large regions of damage that do not fit the profile of thin cracks and patches from previous repairs. Some of the

images also contain areas where prior damage was identified and patched causing an intensity gradient in the images not caused by cracks.

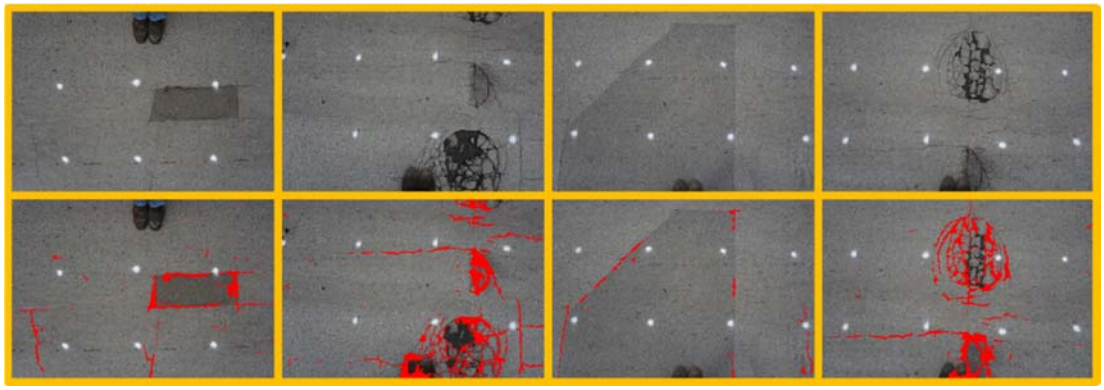


Figure 59. Sample of crack identification with acceptable results due to environmental conditions. Top: Original image. Bottom: identified cracks

Finally, Figure 60 shows some of the worst results from the test which usually occurred at the edges of the structure. Some of the images have a different texture than the rest of the deck. Others have significantly different color caused by a wet surface.

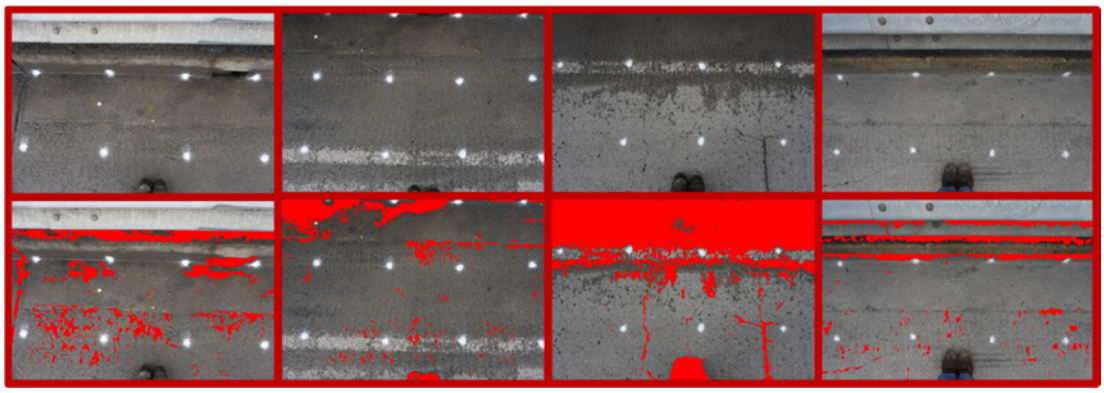


Figure 60. Sample of crack identification with bad results due to environmental conditions. Top: Original image. Bottom: identified cracks

After the cracks were identified, the same points used to stich the original images together were used to stitch the images with the cracks identified together. This information is very important because it allows decision makers to see where on the structure damage is located as well as the severity of damage.

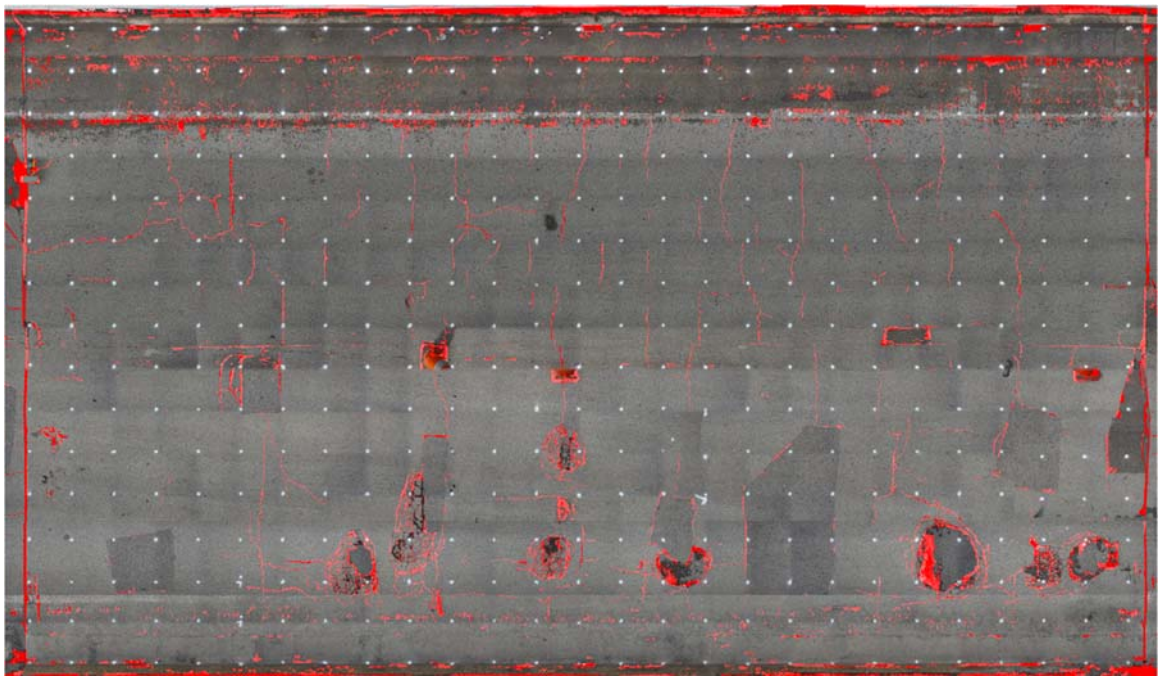


Figure 61. Stitched image with damage information overlaid on the image

Since manual stitching using the known global position of each white paint dot takes a significant amount of time, SFM, explained in section 2.4.6, was also used to stitch the images together. A flowchart for the identification of cracks and generation of a global crack map is shown in Figure 62. The method is more automated than manual point selection for stitching.

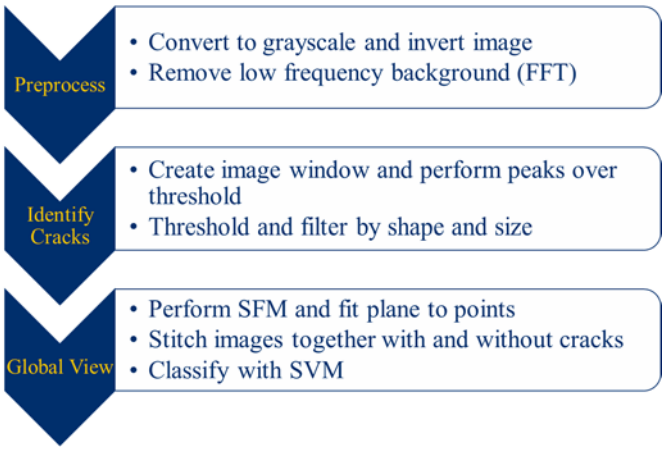


Figure 62. Steps to crack identification, stitching, and classification.

A point cloud was generated of the structure and shown in Figure 63 [176].

Figure 63. Point cloud of the bridge deck [120, 138, 176]

Since the bridge deck is roughly planar, a plane was fit to all of the points and the images were projected onto that plane and stitched together as shown in Figure 64. It is clear from the figure that there are some inaccuracies in the stitching due to residual lens distortion effects and potentially a mismatch in the plane compared to the actual structure. This method still has the major benefit of not requiring the manual clicking of points to

stitch the image. The method provides the same benefits to supplement other NDE technologies with less effort on the part of the user. The cracks can still be overlaid of the structure as they were in Figure 61, but are not presented because the same images were used with a different projection matrix.

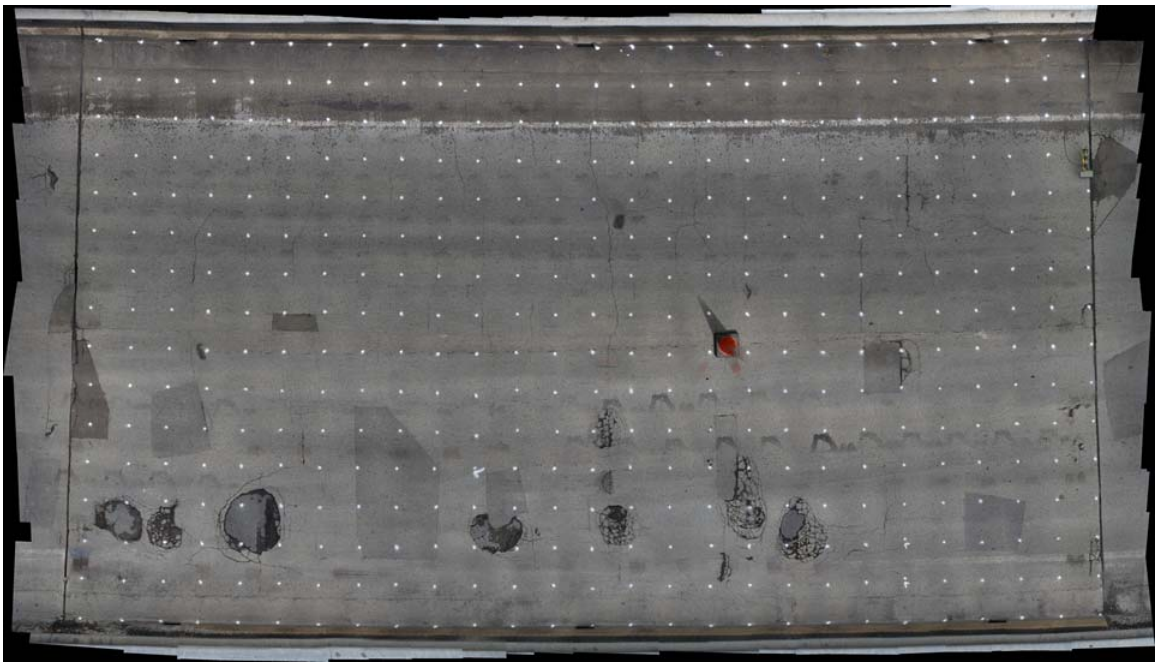
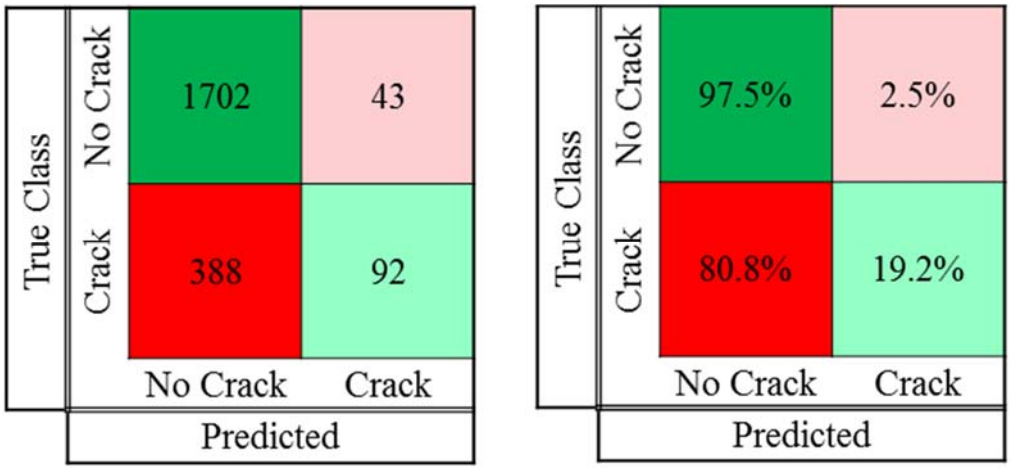


Figure 64. Stitched image resulting from SFM [120, 138, 176]

The next step for crack identification is to use machine learning to aid in the identification of cracks. There were many false positives in the data that make it difficult to accept the results of the initial identifications. Since cracks are generally long and thin, different features were used to classify each blob that was identified as a crack. Roundness, convexity, adjusted roundness, aspect ratio, equivalent diameter, and eccentricity, described in section 2.3.6, were used as features for this classification. A support vector machine was used to classify the cracks. A set of 40 images were used for training the

dataset. Two classes were used: crack and no crack. The results of the training are shown in Table 5 with the number of cracks and non-cracks identified and the percentages of true positive, true negative, false positive and false negative. The overall accuracy of the training was 80.8%, but there is a big skew towards accuracy of the non-crack case. The crack identification was poor based on the training, but this is most likely due to the fact that there were only two classes and the patches produced blobs that look like cracks. In addition, a skew could potentially be added to the false alarm and missed detection rates to optimize them based on the cost of a missed detection and the cost of a false alarm. A higher number of features could also be used to obtain better results as well.

Table 5. Training results [120]



Even though the crack identification with SVM seemed poor based on the table, the results were very effective at eliminating false cracks and identifying actual cracks. A few images after identification are shown in Figure 65. It is clear that the machine learning did improve the identification in the images.

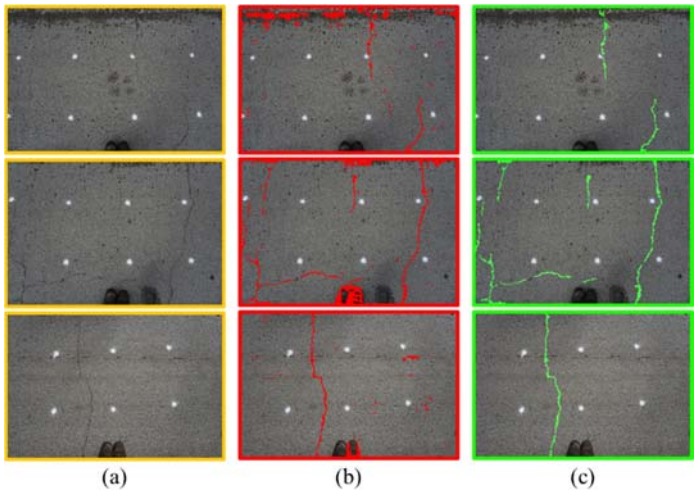


Figure 65. (a) Original images. (b) Cracks identified with peaks over threshold method. (c) Results after SVM classification.

The images were then stitched together with the cracks identified after applying the SVM classification and the full mosaic is shown in Figure 66.

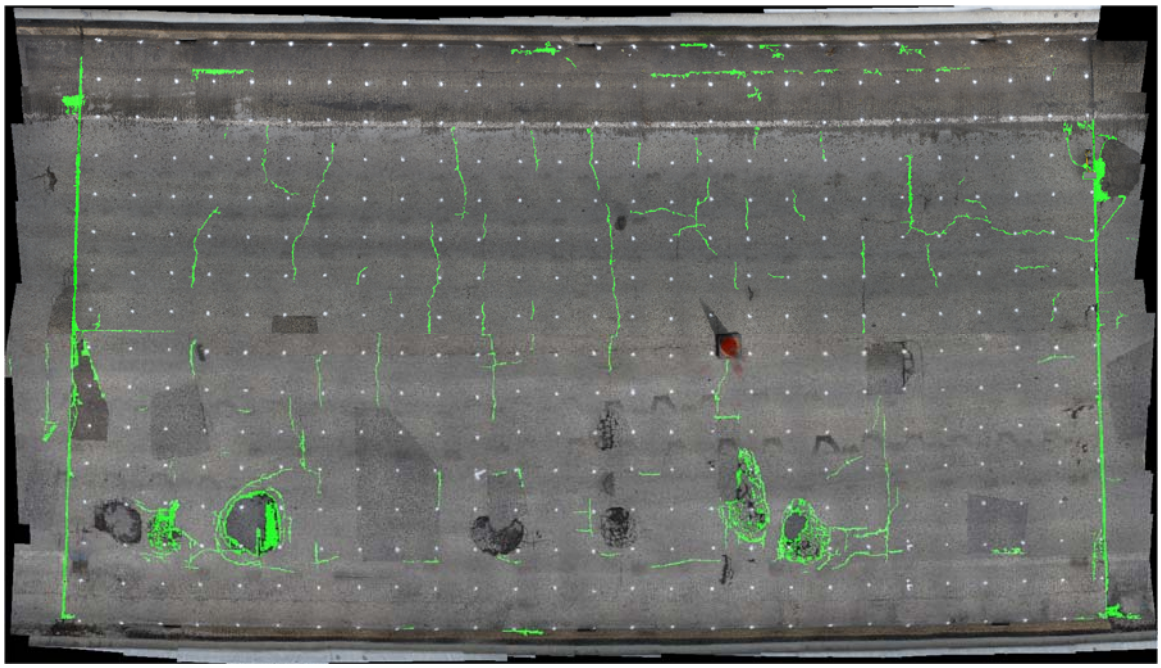


Figure 66. Image mosaic leveraging SFM and SVM classification.

Image segmentation is not a fully solved problem. More complex field images were analyzed with another algorithm tuned for those images. The images were not collected using a UAS, but the locations of the images were not overly controlled when they were taken. These images contained problems that occur in the field including rough surfaces, shadows, traffic, dust, graffiti, image noise, and other operational and environmental factors affecting the quality of the images and reduce the ability of a segmentation algorithm to segment the cracked surface. In the future, more robust and reliable segmentation tools must be implemented to decrease their effect on the results of the identification. The automation of these identification algorithms is crucial to prove the benefits of using UAS as a first pass of inspection. In order to help accomplish this goal, image registration from previous inspections could be used for a direct comparison to determine changes in the structure. Another future objective for crack identification is the use of onboard image processing to provide feedback to the UAS and damage identification algorithm to further analyze the structure with another NDE method. Furthermore, future work will require a better implementation of machine learning with more classes assigned to different types of potential damage. This will limit misclassifications and potentially help to add layers to an analysis tool to aid a user to analyze specific types of damage.

Chapter 5 Subsurface Damage Identification

5.1 Introduction

There has been increasing recognition of the importance of rapid and cost-effective techniques for the assessment of bridge deck condition over the last decade. Bridge decks not only impact on ride quality, but also serve to distribute the traffic loads to the other structural members and protect the superstructure from hostile environmental conditions. The initial cost of a bridge deck is small compared to the other parts of the structure, but they account for between 50% and 80% of bridge maintenance funds [20, 262]. As a result, early deterioration identification is essential to enable preventive and more cost-effective interventions. This will have a significant impact to reduce the life-cycle cost of bridges.

Currently, many different nondestructive evaluation (NDE) methods exist for identifying damage in bridge decks including impact echo, Ground Penetrating Radar (GPR), and several others which require contact with the structure [117]. Some of these techniques have been integrated on a robotic platform and used simultaneously to collect information for a combined evaluation of the bridge deck [20]. Furthermore, non-contact methods for damage identification include multispectral imaging, LiDAR, and DIC to identify both surface and subsurface damage in the case of infrared (IR) imaging [10, 36, 246]. Major benefits of remote sensing include the speed at which data can be collected, the full field nature of such data, and the ease of data interpretation when compared to other methods.

InfraRed (IR) imagery has been used in several NDE applications [200]. For example, infrared images have been utilized to determine the roof moisture content and

analyze the performance of wet insulation [263, 264]. In addition, along with other NDE methods, InfraRed Thermography (IRT) has applications in robotic tunnel inspection [265]. Active thermography requires a heat source or mechanical stimulation to excite the structure to identify defects and has been demonstrated in aerospace NDE applications primarily with honeycomb structures [266]. Both active and passive thermography have been shown to identify damage in bridge elements, such as delaminations in bridge decks [206, 267]. Specifically, detection of subsurface defects in concrete structures is based on the different heat transfer characteristics in flawed regions detected by temperature gradients [268-270]. Heat sources for these applications are typically provided by external heating or solar radiation [271]. In addition, both infrared and color cameras have been used to perform visual inspections using vans, boats, and UAS for top and side views of structures [19, 93, 272]. Geographic Information System (GIS) representations of bridge decks have been produced using infrared images combined with Global Positioning System (GPS) and image stitching tools [273]. Furthermore, IRT has also been used in conjunction with acoustic approaches for monitoring damage on masonry wall structures under cyclic loading [13] and detection of damage in concrete structural components [19].

UAS collected multispectral imagery has been used for many different types of civil applications. For instance, color and infrared images acquired by a UAS were used to identify humans in distress for rescue operations [84]. Furthermore, multiple UAS simultaneously collected data to identify forest fires with color and IR imagery [274]. In addition, UAS collected near infrared (NIR) images have been used to identify crops and segment out other objects like soil [221]. Similarly, IR images were leveraged to determine olive tree health by estimating the chlorophyll content and water distress [85]. Moreover,

the use of multispectral imaging and synthetic aperture radar have been proposed for gas pipeline inspection and permission was given to private contractors to survey pipelines with a fixed wing UAS in the United States [88, 228].

In this chapter, a hexicopter UAS was used as a remote sensing platform to collect IR and color image data and post process the data to detect delaminations as well as to estimate their size in a mock up bridge deck [126]. The delamination locations were not known to the pilot prior testing. The color and IR data was recorded onboard and streamed back to the pilot during the flight which allowed the manual identification of defects in real time, justifying the capability of rapid inspection. The data collected was analyzed after the flight by a novel post-processing algorithm based on the calculation of grayscale gradients and their directions in the IR images to detect and estimate the size of subsurface defects. The results obtained from the UAS imagery were validated by performing measurements with a similar multispectral payload attached to a moving cart to compensate for both payload and UAS motion uncertainties. Furthermore, potential delaminations in the field experiment described in section 4.5 were manually identified using a ground based platform.

5.2 Bridge Deck Description

The bridge deck mockup used to perform these experiments had multiple pre-manufactured defects inside which were unknown to the pilot before, during, and after the tests were conducted [275]. The deck consisted of an eight inch thick reinforced concrete slab with three structural steel I-beams supporting it. Figure 67 shows a top, front, and side view drawing of the structure with dimensions.

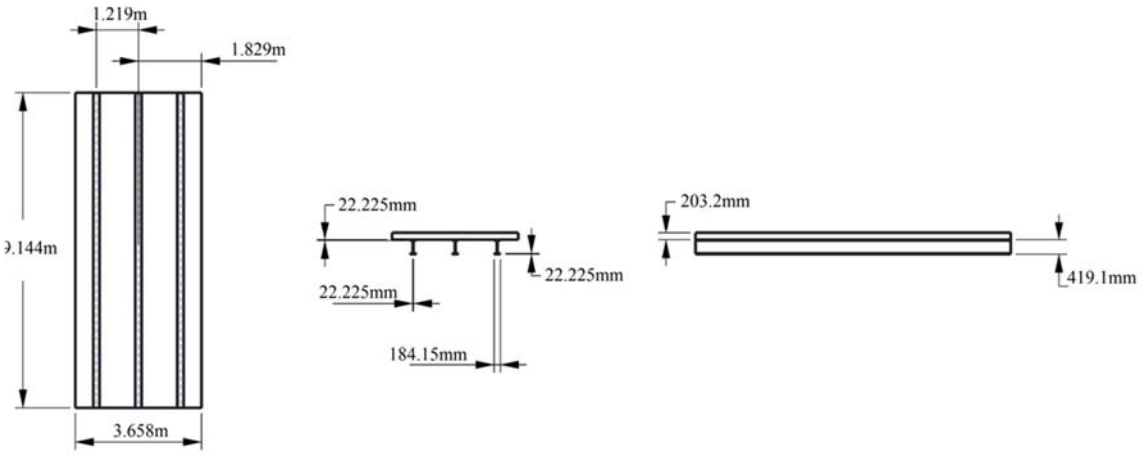


Figure 67. Top, front, and side view of the simulated bridge deck.

The bridge deck width is 610 millimeters smaller than the dimension considered in the ASTM standard [204] which recommends a 4.27 meters minimum field of view for both the color and IR cameras to fully cover an entire lane on actual bridge decks. This was not a concern since the UAS flight plan can be altered prior to the flight to ensure the required field of view is obtained given the properties of the optics of the equipment. It is further expected that with increasing height, the prop wash (turbulence from the propellers) will have less of an effect on the measured temperature of the bridge deck.

5.3 Bridge Deck Delamination Identification and Measurement from UAS imagery without Radiometric IR Data

5.3.1 UAS and Payload

A DJI F550 six-rotor UAS with a NAZA V2 flight controller was used to conduct the aerial experiments. Figure 68 shows the components integrated into the payload and their positions on the UAS. Although the system was equipped with GPS, gyroscope, accelerometer, and pressure sensor for flight control and stabilization, the pilot flew the system manually in stabilize mode to collect the data due to the close proximity of

obstacles. However, a completely GPS controlled flight was also possible using up to 16 GPS waypoints in the flight control software. The UAS carried both a GoPro Hero 3^{+} silver edition color camera and a FLIR Tau 2 uncooled core IR camera to capture video imagery in real time on a digital video recorder (DVR). The FLIR Tau 2 was unable to collect radiometric temperature data and the only output provided was an analog video of surface temperature gradients. At first, the color camera was placed on a gimbal to keep it level with the ground at all times; however due to the weight limitations on the gimbal, the IR camera was fixed to the bottom of the UAV with a separate vibration dampening system. Difficulties in image registration between the color and IR data resulted in more testing with both cameras fixed to the bottom of the UAS to keep a constant transformation between the cameras. This transformation was calculated with an over determined direct linear transformation or homography matrix described in section 2.4.4 [143]. Due the orientations of the cameras installed on the UAS, the transformation was assumed to have scaling, rotation about the axis perpendicular to the image plane, and rotation within that plane. Therefore, only two corresponding points were required to calculate the similarity transformation, but more points achieve better accuracy by using the least squares approach shown in equation (2.21). These corresponding points were manually selected in 10 sets of corresponding images to ensure the transformation was accurate. The calculated transformation had negligible rotation and the standard deviation associated with the translation was 4 pixels in the horizontal direction and 5 pixels in the vertical direction.

The color video was recorded using the memory card inside the camera and the IR video was saved onto a second memory device using an onboard digital video recorder (DVR) shown in Figure 68. The color video had a 138° diagonal field of view, resolution

of 1920x1080 pixels, and recorded at 30 frames per second. The IR camera had a 69° diagonal field of view, resolution of 324x256 pixels, and also recorded at 30 frames per second. The clocks of the two cameras were not synchronized during the flight, but the resulting images were post processed to be synchronized manually.

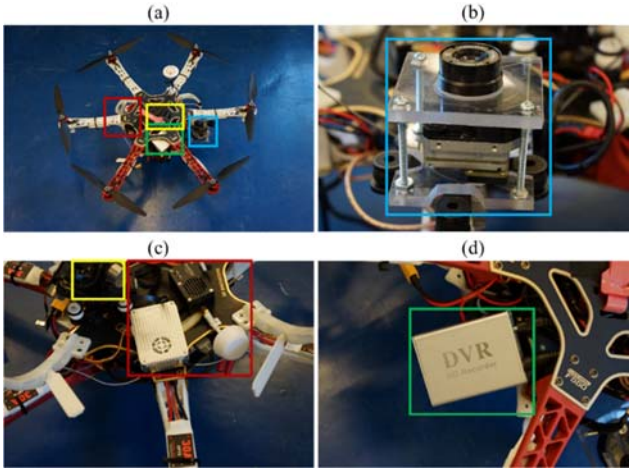


Figure 68 (a) UAS platform with data capture equipment highlighted with the colored boxes. (b) FLIR Tau 2 with vibration dampening system (c) Antennas for RGB and IR video streaming (red) and RGB camera and gimbal (yellow). (d) Onboard digital video recorder for saving analog IR video.

The ground sample distance varied with the height of the cameras, which was not recorded during the tests due to the closed system in the flight controller. This was not in issue for identification, but for measurement, a reference in each image was required to get an accurate estimate in damage size. The range of ground sample distance varied roughly between 2.5mm (at 2m height) and 1.3 cm (at 10m height) throughout the test. Both videos were wirelessly streamed to the pilot using 2.4GHz for the high definition DJI Lightbridge color video and a DJI AVL58 5.8GHz video downlink for the standard definition IR video. Figure 69 (a) shows an example of an IR image overlaid on the RGB image after the test

with the areas of potential interest manually identified in yellow boxes. The UAS videos were streamed separately back to the pilot and used to change the flight pattern based visual identification of areas of interest through their observation in real time as well as aid the pilot in navigation.

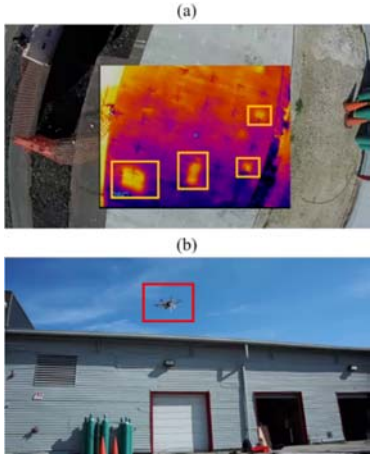


Figure 69. (a) Images streamed in real time from the UAS. The delaminations were manually identified, but are viewed real time from the video stream. (b) UAS hovering over the location of the delaminations in (a)

Several flight tests were completed prior to testing on the simulated bridge deck. The first tests recorded data on the ground to reduce the weight of the aerial system and increase the flight time. However, during these tests, it was observed that when the videos were recorded on the ground, there were missing frames due to streaming. This was mitigated by wiring the recording systems on the UAS to ensure that no frames of the video were missing due to radio frequency interference caused by possible obstacles between the system and the ground station. Both videos were still projected real-time onto screens so the pilot could see what the UAS was capturing real time. A diagram of the data flow and an image of the ground station and UAS during a flight are shown in Figure 70. The

receiving antennas were placed outside of the van for the experiments to limit the problem of shielding which cause breaks and jumps in the video streams.

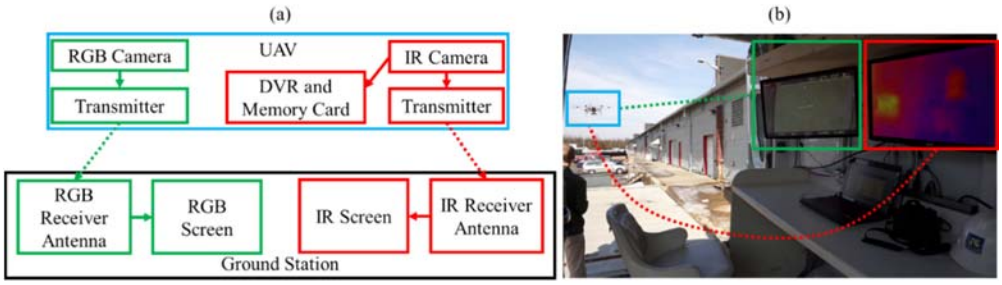


Figure 70. (a) Data flow diagram. (b) Image of ground station and UAS during flight highlighting the data obtained from the RGB and IR camera and streamed to the ground.

Due to the number of video components and the power drawn by these systems, a second battery was added to the system to power the video components. A diagram of the components powered by the added battery are shown in Figure 71. A 3S battery was chosen because it has a lower voltage and low weight. Since each system did not draw much power compared to the motors, the UAS battery was expected to be out of power well before the second battery.

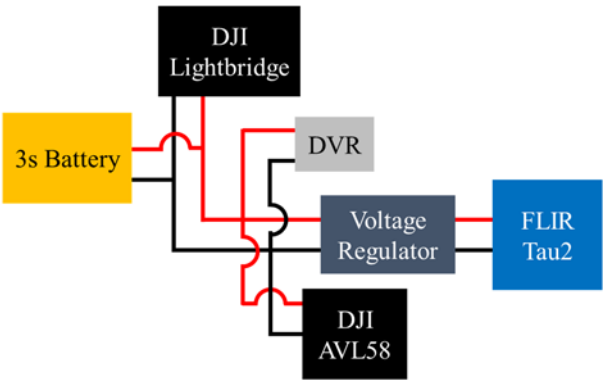


Figure 71. Power system for the video components on the UAS

The total takeoff weight of the system was 2.6 kilograms which is close to the maximum recommended takeoff weight for this UAS. Despite this added weight, the system achieved flight times of up to 7.5 minutes in a very low wind environment. This was achieved by placing different components in strategic places to reduce the effect of moments on the aircraft to keep the center of mass near the center of the aircraft. However, when there was more wind, the control of the aircraft degraded, so a larger UAS would be needed in order to maintain good control and carry the payload.

5.3.2 Validation Equipment

A ground based platform including both infrared and color cameras was used to validate the results obtained by the UAS. A GoPro and a FLIR a325sc were attached to a rolling cart and moved along the deck to identify the delaminations with the color imagery that served to help localize the infrared data. Figure 72 shows the setup of the FLIR (red), GoPro (yellow), power (green), and recording system (blue) contained on a cart. Since a generator was used to supply power, the whole system could be moved easily without connection or range problems due to power cords which is similar to what must be done in the field. This allowed for ease of motion and therefore smoother movements which made the data easier to process.

The FLIR a325sc was able to provide thermography data (actual surface temperature values) as opposed to the Tau 2 used on the UAS. This simplified the post-processing because the FLIR a325sc temperature images were not affected by the color changes imposed by the automatic gain control (AGC) and allowed the processing do be completed directly on the temperature values. The ground based setup was also more controlled because the working distance of the cameras did not change with respect to the

deck throughout the test and the location was easily controlled. This differed from the UAS which was manually controlled in stabilize mode using only feedback from the IMU and pressure sensor, so a variability in position and height was expected, especially during wind gusts. Wind was not a significant problem for camera movement with the ground setup. Vibrations were caused by the generator, but they had a negligible effect on the images due to the dampers on the generator, dampening within the cart, and the shutter speed of the cameras. Furthermore, the images were sufficiently clear to manually identify the delaminations in the real time videos.

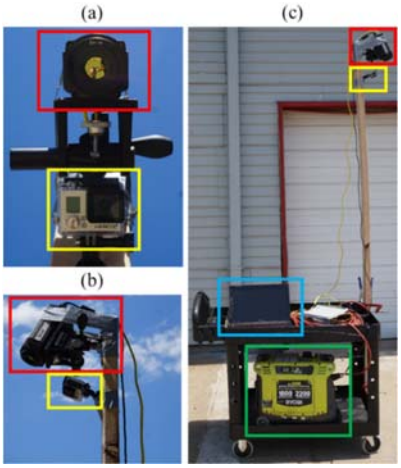


Figure 72. (a) and (b) show the FLIR a325sc (red) and GoPro camera (yellow). (c) Shows the full ground system including the cart with the generator (green) and laptop (blue) used for data collection.

5.3.3 Algorithms and Image Analysis

5.3.4 UAS Delamination Identification Algorithm

The pseudocode developed for delamination identification using color and IR images extracted from the IR analog video is shown below.

Load color and IR images
Obtain same field of view for color and IR images
    Correct images for lens distortion
    Project color image onto IR image
Identify the deck in the color image
    Mask the IR image keeping pixels identified as the deck in the color image
    Convert IR image to grayscale
Identify seed points
    Calculate the gradient of the grayscale IR image
        Threshold the gradient based on Otsu method and mask IR image
        Use local histogram equalization on the rest of the IR image
        Color threshold using Otsu method
    Calculate extrema of each blob
For each blob
If the maximum of the gradient is greater than threshold
Use top left blob extrema as seed point
Use region growing with 2 standard deviation from the mean cutoff criteria
End if
End for
For each blob
    While current pixel is unanalyzed and less than specified distance from the blob extrema
        Calculate the gradient and the normal of the gradient
        Use top left blob extrema as starting point
        If the magnitude of the gradient is greater than the mean plus one standard deviation
            Follow gradient direction to the next pixel
        Else if the magnitude of the gradient is less than the mean minus one standard deviation
            Follow the opposite direction of the gradient to the next pixel
        Else
            Follow the normal direction to the next pixel
        End if
        If next pixel has been analyzed
            Blob is a delamination
        End if
    End while
End for

Display results

Both color and infrared images were loaded and corrected for lens distortion. The equations used to model the lens distortion in the image are shown in equations (2.17) and (2.18). Three radial distortion parameters and two tangential parameters were used to correct the distortion in these images. The same calibration described in section 4.2 was used to correct the distortion in the GoPro images. For the IR images, it was more difficult to obtain high quality images for the calibration process due to the lack of contrast in the calibration images. To this aim, a checkerboard pattern was printed on a projector transparency and heat was applied using lamps to achieve higher contrast than the paper used for the color image lens distortion calibration [276]. An example of one of these images is shown in Figure 73. The heat sources are boxed in red and the checkerboard used for calibration is boxed in blue.

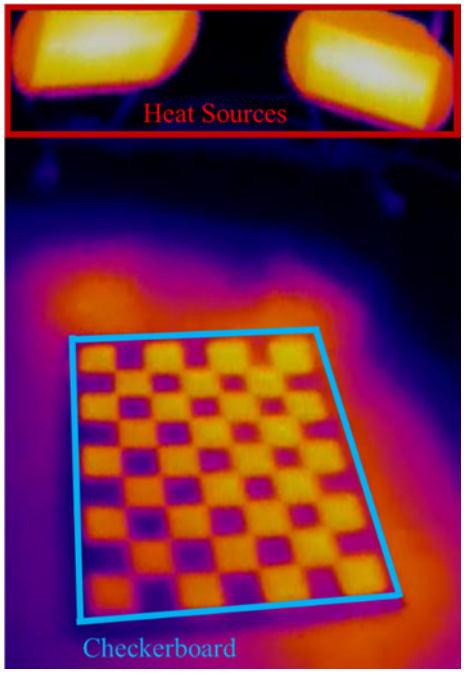


Figure 73. IR picture used for calibration

The heat source created higher contrast, but the edges and corners of the checkerboard were still not as well defined in the images as the color images. Therefore, 50 images were used in the calibration in hopes that the error minimization in the camera calibration algorithm would reduce the error of the model by minimizing the error associated with the corner identification. The error associated with reprojection was two pixels after the minimization and visually, the lens distortion was improved significantly. Localization of the checkerboard corners was determined to be the biggest source of error due to the lack of clear corners. The pixel error associated with the IR images was expected to affect the results of the algorithm more significantly than the color imagery. Delamination identification performance is not affected by lens distortion errors, however the measurement steps will be affected up to 4 pixels the horizontal and vertical directions in the images. All of the errors associated with camera calibration are reported in pixels since the physical dimension associated with the error is dependent on the working distance of the camera to the specimen. The error can be calculated using equation (5.1) where θ is half of the angular field of view, h is the height of the UAS, N_{pix} number of pixels across the angular field of view, e_{pix} is the error in pixels, and e is the error in dimensional units. A diagram of the locations of these variables in the measurement system are shown in Figure 74.

$$e = \frac{h \tan(\theta)}{N_{pix}} e_{pix} \tag{5.1}$$

An example for calculating the error bounds on a specified size of a delamination is shown below using the error values identified above. Assuming a 0.305m x 0.305m delamination is identified perfectly in an image taken perpendicular to the deck at an altitude of 4m (h), the 4 pixel error (e_{pix}) and 69º field of view (2θ), the worst case would

identify the lower bound at 0.251m x 0.251m and upper bound at 0.358m x 0.358m. It is expected that the results should be significantly better since these errors are based on a 3 standard deviation reprojection and the large pixel error comes from poor localization of the corners in the IR images.

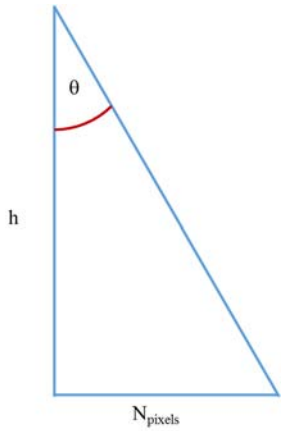


Figure 74. Diagram of working distance (h)

The color image was set to have the same field of view as the IR image using a projective transform calculated with several corresponding points that were in both the color and infrared images. Since there was a constant distance and angle between the cameras, the same transformation could be used for all of the images obtained during the test. The transformation was calculated multiple times with different images using at least four matching tie points in pairs of images. Since the field of view of the color camera was much larger than that of the IR camera, the color image was cropped to remove the edges outside the field of view of the IR image. This also served to eliminate the areas of the image that had higher amounts of distortion remaining after the correction. Furthermore, since the color image was not used for detection purposes, bilinear interpolation was used

to determine the color of the pixels to complete the transformation. After the transformation, the deck was determined by color and a mask was formed for the image to eliminate the background. Since the images were aligned, the same mask created to identify the deck in the color image was used to mask the IR image. The hue, saturation, intensity (HSI) color space was used to identify the deck because the color based thresholding generally performs better in this space as explained in section 2.4.1 [118].

After the deck was identified, a threshold based on the gradient values was utilized to mask the IR image [118]. In addition, local histogram equalization [118] was used on the grayscale IR image to enhance the contrast in the images of irregular portions of the deck. Furthermore, the same method served to mitigate the varying colors in the images caused by different light conditions and the automatic gain control of the camera. Since the automatic gain control changes the color of the images based on the scene to create the best contrast, a constant threshold value across all of the IR images could not be used. Therefore, Otsu's method was chosen to segment the IR images [121].

The measurements occurred early in the day and consequently the delaminated areas corresponded to warmer regions on the deck surface. Therefore, these areas appeared brighter in the IR images [204]. The images were converted to binary and the extrema of each blob was used as the seed point for the region growing algorithm [118]. The criteria used for region growing was based on the first and second moments of the pixel values in the blob at each iteration [110]. The lower bound for acceptance of a pixel into the region labeled as a delamination was set to the average minus one standard deviation. There was no upper bound because the delaminations were expected to brighter than the rest of the deck because the delaminations were expected to be hotter. An upper bound would have to

be added if there were other hotter objects on the deck that would make the delaminations appear cooler in the image. Since the IR camera used did not keep constant colors throughout the image set, the color was not a reliable identification feature and therefore color gradients were required to help identify the delaminations. The extrema of each blob (as defined by the MATLAB function regionprops) were used as starting points for the gradient following algorithm [277]. The normal of each gradient was calculated and if the magnitude of the gradient was within a standard deviation from the average gradient for that blob, the normal was followed. If the gradient magnitude of that pixel exceeded the upper bound, the direction of the gradient was followed to select the next pixel. If the magnitude of the gradient was below the lower bound, the opposite direction of the gradient was followed to select the next pixel [131]. Since delaminations generally do not typically appear as shapes with sharp corners in the IR images due to the heat transfer properties through materials and they are large enough in size to form a set of gradients encompassing the delamination, any shape enclosed by the set of pixels identified by the algorithm was assumed to be a delamination. If the pixels following the gradient approach does not enclose a shape, the blob was not identified as a delamination. This made it impossible to correctly identify delaminations that were not fully contained within one image. In the case that the pixel being analyzed fell outside the image, the algorithm would stop causing a missed identification if that blob was in fact a delamination. This was not a concern due to the large field of view of the FLIR Tau 2 and the number of images that covered the entire deck in multiple images.

After detection, size estimation was achieved using the width of the bridge deck in images that contained the entire width of the bridge deck. For those images that were taken

at lower altitudes and therefore did not contain the entire deck width in the field of view, such measurements were not possible. The edges of the deck were calculated using the canny edge detection method and Hough transform to identify the line segments making up the edges of deck. The distance per pixel in each image was calculated using the known width of the deck and calculating the number of pixels between the parallel lines. The assumptions made were that the deck edges are parallel and the UAS very close to level at the time the images were captured. The pitch and roll of the vehicle are therefore critical in the results of the algorithm so when wind causes the UAS to rotate at a slight angle, the distance per pixel would vary and the estimated delamination sizes would result in larger errors. Equation (5.2) was used to estimate the error associated with non-level flight. In the above expression, the camera calibration matrix (K) was assumed to be constant because the same camera was used for all of the images, R is the rotation matrix, the vector $[X, Y, Z, 1]^T$ represents the distance between camera and observed delamination, and $[u, v, 1]^T$ represents the pixel coordinates in the image [143].

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{5.2}$$

Since the pitch and roll angles are small while the system is hovering, only angles of less than 5 degrees were considered. The height of the system was not constant during the test flights so test cases of 2 meters and 10 meters were used as extremes. A target delamination size of 0.305m x 0.305m was considered since this is a common size that can be identified in the field. The rotation matrix (R) assumed that pitch and roll varied of the same angle value, and the position of the delamination from the camera ($[X, Y, Z, 1]^T$) was

assumed to be known to demonstrate the error. The changes in the pixel coordinates were analyzed using the ground sample distance at the given altitude.

As expected, as the angle error was increased, the error in the measurement also increased. The error also increased with increasing height; however the acceptable error associated with the pitch and roll attitude of the aircraft increased from 0.3 degrees at 2m to 3 degrees at 10m based on the ground sample distance. A potential solution to help mitigate this problem is the use of a gimbal that is capable of carrying both cameras to keep them facing down, but this requires a UAV with higher payload. The yellow and white traffic lines on roadways could also serve as reference in future tests on an actual roadway.

5.3.5 FLIR a325sc Delamination Detection Algorithm

The algorithm previously described for UAS imagery was adapted for delamination identification using the ground IRT data. This data has temperature values associated with each pixel which is not the case for the FLIR Tau 2 images. Therefore, local histogram equalization and optimal thresholding were unnecessary because these the main purpose of these two steps was to even out the data in the images so that there was a more constant effect throughout them. Since the base temperature of the undamaged deck did not change within a single frame, histogram equalization was redundant and would actually cause a loss of information. The mean and variance of the thermography data and the temperature of the undamaged deck were used as the first iteration of segmentation instead of optimal thresholding with the Otsu method. This step does assume that the temperature of the deck does not change between frames which was necessary in the case where a cloud would block the sun for short periods of time. The region growing and gradient following acceptance criteria were altered due to the low variance of the temperature data. The narrow

field of view made the color images unnecessary because the IR data only contained the deck so segmentation of the deck was not needed. Furthermore, since the camera was fixed at 2.57m from the bridge deck and the field of view was so narrow, the size estimation could rely only on the location of the camera instead of a reference scale in the images.

5.3.6 Results and Discussion

The overall image of the tested mock-up bridge deck is shown below in Figure 75. The color and IR images were taken to display the likely locations of the areas of interest in the infrared image and map their approximate location in the color image.

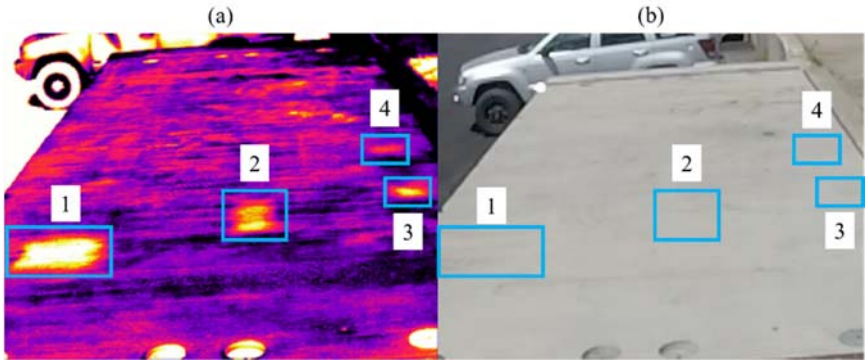


Figure 75. (a) Infrared image of the entire deck obtained with FLIR a325sc camera. (b) Color image of the entire deck.

The images captured for this experiment were extracted from videos. As a result, over 4,000 images were analyzed in each aerial test. The measurement step was only applied to images that contained the full width of the deck in the field of view. Images that were obtained from different flights performed during different days were analyzed. The first flight used a gimbal to hold the color camera and therefore the transformation between cameras could not be assumed constant and therefore was not as effective for deck segmentation. A small sample of the results from these images are shown in Figure 76.

The height of the UAS was not precisely controlled or recorded and due to the nature of manual, piloted flight in stabilize mode, the images had different fields of view. If the UAS turned or was moved by the wind, the color camera would stay level and move smoothly because it separately controlled with the gimbal while the IR camera would not because it was fixed to the UAS. This had an effect on the identification algorithm which is clearly observed in Figure 76 (j) and (k) where the identified areas are misaligned and the edge of the deck is identified as a delamination. In fact, Figure 76 shows that all of the shallow delaminations as well as some surface roughness were properly identified based on an independent IR survey conducted with a more sophisticated infrared camera installed on the rolling platform. Figure 76 (l) represents one of the worst results in this scenario in which the majority of the identified areas were due to the surface roughness of the deck.

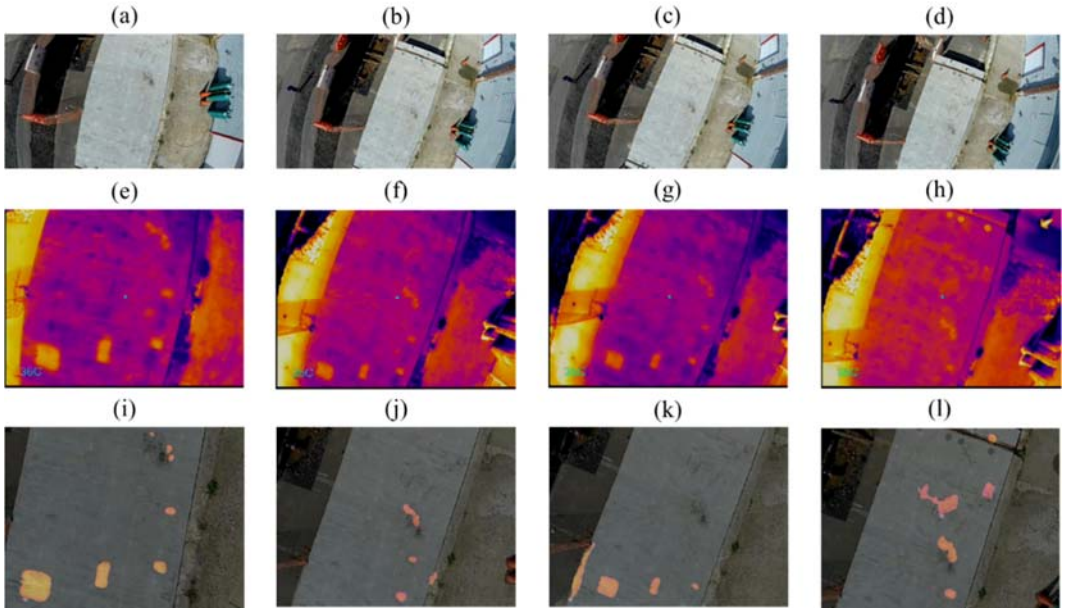


Figure 76. (a-d) Original color imagery extracted from color video. (e-h) Original IR images corresponding to the color images (a-d) extracted from FLIR Tau 2 video. (i-l) Identified delaminations using the described algorithm.

A second flight was conducted to mitigate the issues that arose in the first flight and samples of images from the second flight are shown in Figure 77. The payload for this flight had both the color and IR cameras fixed to the bottom of the UAS to keep the transformation between the color and IR images throughout the flight constant. The orientation of the cameras was changed to shift the center of gravity closer to the center of the UAS for better stability and distribution of weight to increase the flight time. This did not have an effect on the identification algorithm with the exception of a different transformation used to relate the two data sets. The fixed transformation between the color and IR images did improve the results because the deck was easier to identify which allowed for better segmentation of the deck from the color images and therefore allowed the application of the size estimation step which had not been possible in the previous flight data.

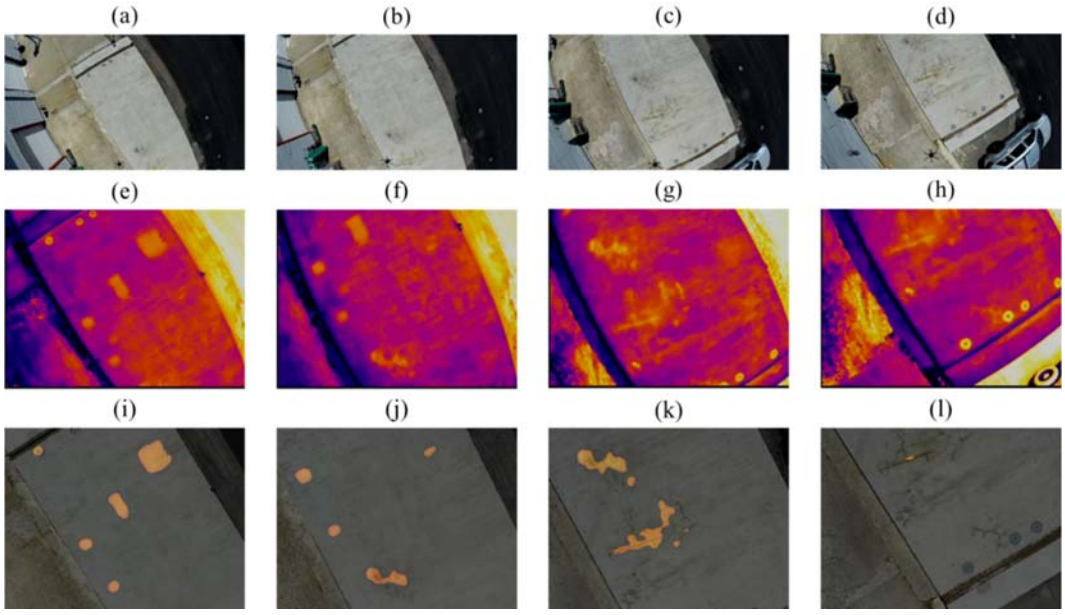


Figure 77. (a-d) Original color imagery extracted from color video. (e-h) Original IR images corresponding to the color images (a-d) extracted from FLIR Tau 2 video. (i-l) Identified delaminations using the described algorithm.

In Figure 77(i), the algorithm correctly identified delaminations in the image in addition to one section at the top left of the image that is caused by a change of the material with different thermal emissivity values causing apparent temperature variations which was verified by the color camera. As expected, surface roughness and different materials on the deck that were in the field of view of the IR images were falsely identified as delaminations. This is displayed in Figure 77 (k), but it also occurs in (j) along with the two delaminations that were correctly identified. The cracks on the surface of the concrete showed up bright in the IR image due to the automatic gain control of the FLIR Tau 2 which optimized the image quality for contrast. In Figure 77(l), one of the blobs falsely identified in Figure 77(k) was limited in size due to the change in the automatic gain control. This significant change in color with a small change in temperature affecting the FLIR Tau 2 images limits the potential of this detection algorithm. However, it can still identify the delaminations with few false positives caused by surface defects.

The redesigned algorithm was applied on the FLIR a325sc images and the results are shown in Figure 78. The IR images in Figure 78 were created using the minimum and maximum temperature to normalize the data into an image for visualization purposes. The actual thermography data was directly analyzed by the algorithm. Figure 78 (i) and (j) demonstrate that the delaminations were properly identified based on visual and manual interpretation of the temperature data obtained with the camera. The pixels representing delaminations appear to be a better estimate the size of the delamination than that of the FLIR Tau 2 images. This is likely due to the different magnitudes of gradients and values associated with the image processed by the automatic gain control versus the thermography data. The apparent delamination in Figure 78(k) is caused by the surface damage on the

deck. The cracks that were falsely identified in the Tau 2 images were also identified when the thermography data from the FLIR a325sc was used. These surface anomalies resulted in fewer erroneously identified delaminations suggesting that the algorithm can be more effective when leveraging temperature measurements than by directly using color images obtained with an IR camera.

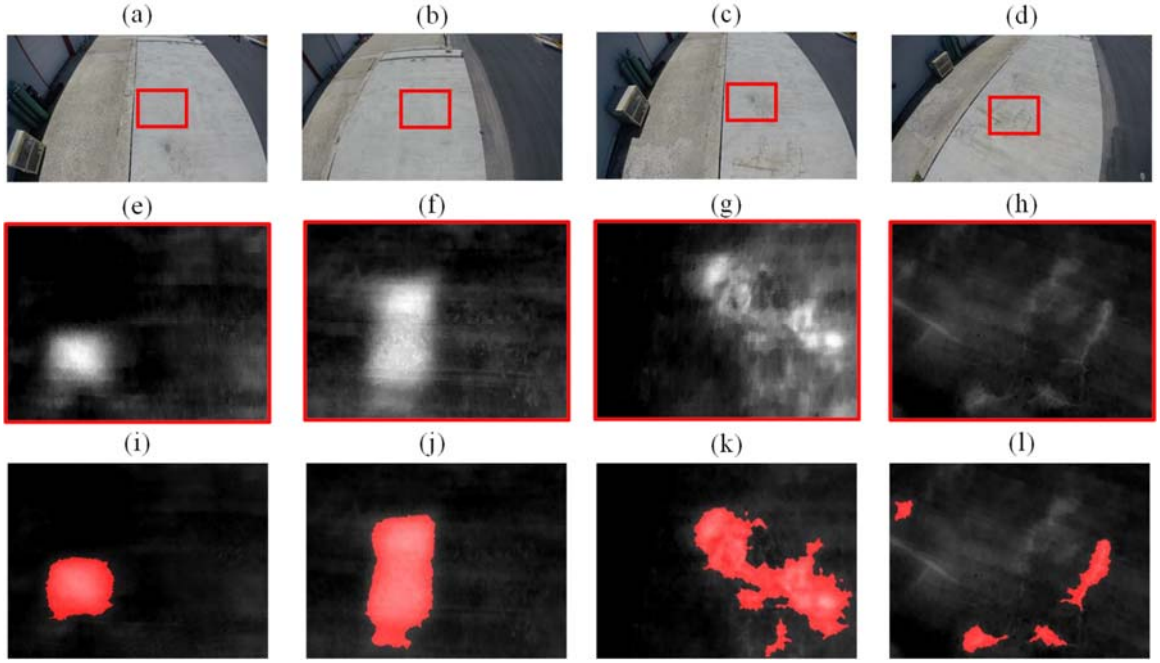


Figure 78. (a-d) Images extracted from color video. (e-h) Processed IR thermography data corresponding to the color images (a-d) extracted from the FLIR a325sc. (i-l) Areas identified as delaminations using the described algorithm.

The delaminations labeled 1 through 4 in Figure 85 correspond to the labels in Table 6 and Table 7, which show the estimated size of each delamination for each measurement system. The dashed line shows a potential delamination that was not consistently identified by the algorithm, but is slightly more visible when looking at the

pictures. The upper and lower bounds are based on the student t-distribution with the degrees of freedom associated with the number of images each delamination appeared in.

Table 6. Estimate of delamination size from UAS aerial imagery

Delamination	# of Images	Average (ft^2)	Stdev (ft^2)	Upper Bound (ft^2)	Lower Bound (ft^2)
1	96	4.02	0.21	4.43	3.61
2	91	1.95	0.19	2.34	1.57
3	82	0.68	0.08	0.85	0.51
4	44	0.60	0.07	0.74	0.46

The number of images each delamination appeared in was determined by the number of frames showing the delamination in the recorded videos. Less frames were captured with the FLIR a325sc because the test setup was easier to control and also took a shorter amount of time to take fewer images. Delamination 3 has a higher number of frames because it appeared and was identified in significantly more frames than any other delamination due to the way the experiment was conducted.

Table 7. Estimate of delamination size using the FLIR a325sc temperature data

Delamination	# of Images	Average (ft^2)	Stdev (ft^2)	Upper Bound (ft^2)	Lower Bound (ft^2)
1	14	3.04	0.09	3.24	2.85
2	14	2.02	0.11	2.26	1.79
3	45	0.84	0.13	1.11	0.58
4	6	0.57	0.09	0.79	0.34

Comparing the results from both experiments using a t-test assuming different variances and the same mean, delaminations 2 and 4 have the same mean in the aerial and ground based setups with 95% confidence and delaminations 1 and 3 do not. This shows

that the two identification methods are statistically different which is likely due to the difference in type of data, and the slightly altered algorithm for the FLIR a325sc frames.

Finally, the results of the test were sent to the owners of the simulated bridge deck to determine how the identification and measurement algorithms performed. The defects were manufactured into the deck during construction with foam and plastic sheets used to simulate the damage. They were placed in predetermined locations inside the deck and kept secret from anyone performing an experiment to test new NDE equipment. The size and the three dimensional location of each defect was recorded during construction and full photographic documentation of the rebar cage with the defects was available to the owner [278]. The simulated damage that was correctly identified and the associated size were confirmed by the owner leveraging this documentation. The four shallow delaminations (shown in Figure 85 in yellow boxes with continuous lines) are 2 inches deep and were successfully identified by the algorithm and both the location and size were properly estimated. However, there are several delaminations, mostly deep into the concrete deck, that were not identified due to the lack of temperature gradient caused in the IR imagery. Only one shallow delamination (within 2 inches from the surface, showed in Figure 85 in a yellow box with dotted lines) was not correctly identified by the algorithm, although it was observable in several IR video frames. The deck includes additional 5 deep delaminations (at depths between 4 and 6.5 inches) that did not appear in any of the images from either IR camera; however, this is expected since deep delaminations do not significantly influence the temperature of the deck surface [278]. The owners did not disclose the sizes or locations of the deeper delaminations.

Table 8. Actual delamination size and average IR measurement error for UAV and ground based setup

Delamination number	Actual			UAV Average Error	Ground Platform Average Error
	Length (ft)	Width (ft)	Area (ft^2)		
1	2	2	4	0.5%	-24.0%
2	2	1	2	-2.5%	1.0%
3	1	1	1	-32.0%	-16.0%
4	1	1	1	-40.0%	-43.0%

Table 8 shows the comparison of the data obtained during the test with the infrared cameras and the information provided by the deck owner. The size estimates had relatively large amounts of error, especially for the smaller delaminations. This is caused by the dissipation of heat and the gradient it causes on the deck surface across the delamination. The threshold for identification for the ground based setup could be changed to improve the results, however an increase in sensitivity will also result in false positives. Overall, it could be concluded that it is difficult with the current setup and algorithms to determine accurately the exact size of the delaminations due to the amount of variability in the environmental conditions. However, it could be further concluded that the presented method can be successfully used to rapidly identify shallow delaminated areas that are the most common in bridge decks and guide more expensive ground-based NDE systems to perform further assessments. It can also be concluded that the relative size of the delamination can be estimated and achieve accurate enough results to determine what areas of the deck are worse than others.

5.4 Bridge Deck Damage Identification with UAS collected Radiometric IR Data

5.4.1 Equipment

A DJI F550 frame was used with the similar components to the system described in section 5.3.1, but the flight controller was a Pixhawk running the PX4 flight stack. The UAS was set to have a combination of two color cameras and an ICI 9320 infrared camera that can record radiometric temperature measurements. The camera had a fixed focal length of 13mm and was light enough to be carried on the UAS. The UAS had a digital video recorder (DVR) and a memory card to save the analog video of the infrared camera while the temperature measurements were stored on a flash drive connected to the UAV module that controlled the camera. The downside to this camera was that it could only record temperature data at a maximum of one frame per second which is why the video recording was required. It did not provide temperature measurements, but it was able to save the data much faster. A Canon PowerShot ELPH 130 IS color camera was used for the collection of images at the same time as the infrared images were taken. The cameras were remotely triggered by the pilot during the test through a switch mapped to an output port of the Pixhawk. The Canon PowerShot had the CHDK installed on it and ran a program that enabled the camera to take an image every time it received a 5V input through the USB port [226]. The other color camera was a GoPro Hero 3^{+} which was used to collect video. The color data from both cameras were saved on their respective memory cards. The color and infrared cameras were fixed to the bottom of the UAS to ensure the distance between them did not change through the duration of the flight for the reasons explained in 5.3.1 [14]. The data capture schematic is shown below in Figure 79.

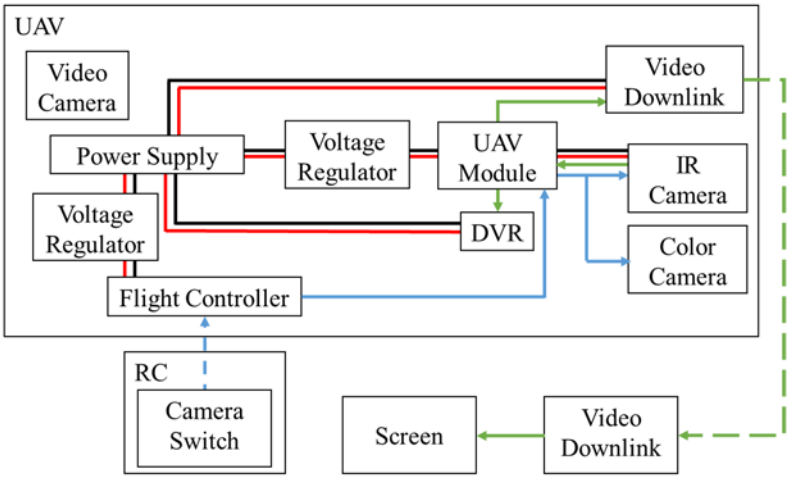


Figure 79. UAS data capture schematic

The red and black lines indicate the power for each component of the data collection system. Any subsystem not connected to the main power had an internal battery that powered only that component. The green lines are the image and video data captured from the infrared camera. The UAS module received video from the IR camera and send it to the DVR to record it. The IR data was saved by the UAS module simultaneously on a flash drive when the camera was triggered. The analog video was also sent to a transmitter which steamed the video real time to the ground. The blue lines represent the flow of the control signals to the data collection system. On the remote control used to fly the UAS, a switch was mapped to a port on the flight controller to trigger the color and infrared cameras that would not hinder the pilot's ability to fly the aircraft. In a scenario where multiple pilots are employed to complete the flight, a second controller could be used to control the data capture. The goal of this setup was to obtain images in color and infrared data at the same time. The color and infrared videos were saved on the DVR and the internal storage of the GoPro for visualization and documenting purposes to help clarify the movement of the UAS after the test if it was required.

A color and infrared camera were placed on a rolling platform and images were taken in the same location with each camera to have better control of the position of the cameras at the time of image capture [19, 279]. The infrared camera was the same 320x240 pixel ICI 9320. The color camera was a 24 megapixel Sony a6000, with a 50mm focal length. The focal lengths were chosen so that both cameras had the same field of view based on the sensor size and focal length. It should be noted that the field of view is based on the focal length and the sensor size which is why the focal lengths are different and achieve the same field of view.

5.4.2 Damage Identification

The color images were post-processed in MATLAB to identify cracks. The first step was to sharpen the image using an unsharp mask [118] and then threshold the image based on the mean intensity. The image was resampled to become smaller and the process was repeated for four different sizes of the image to reduce the noise [280]. The segmented parts of the image were then filtered by shape using the roundness criteria from equation (2.11) to eliminate regions that appeared to be round. The locations of these identified cracks were saved and overlaid on the original image.

The delaminations were identified using the infrared imagery obtained during the test. The temperature values in the infrared image were segmented using a threshold value based on the mean temperature value within the image or an image set. Using an image set to determine the average temperature is beneficial to reduce the problems of a delamination taking up a large portion of the image and it also serves as a way to decrease the effects of temperature changes associated with environmental conditions such as clouds. Another benefit of using multiple images is the ability to use the standard deviation to identify the

delaminations as well. If the standard deviation is greater than 1, a threshold was set to the sum of the mean and standard deviation. If the standard deviation was less than 1, the threshold was set to the mean plus 0.5 degrees based on the ASTM standard for infrared thermography in bridge decks [204]. The gradient was calculated using the Sobel operator as a second metric to determine if a delamination was present. Any area of the image that satisfied both criteria for temperature value and temperature gradient was flagged as an area of interest and therefore a potential delamination. The locations were saved and overlaid on the infrared images.

After the areas of interest were identified in both sets of images, the infrared image was mapped onto the same field of view as the color image using the same method described in section 5.3.1. This simplified the processing significantly because this fixed transformation allowed for the keypoints identified in the color images to be used in the infrared images as well for stitching purposes. The surface and subsurface damage information were projected onto the same image to display all of the local information associated with that camera location. The color images were used for display purposes because the infrared images consist of temperature values with colors assigned to them to achieve the best visualization. The temperatures were rescaled for visualization purposes, but did not have any meaning after the identification was completed. The color images also had higher resolution and are therefore a better choice for displaying all of the post-processed damage information. A diagram of the post-processing algorithm is shown in Figure 80. The yellow segments correspond to the color imagery, the red segments correspond to the infrared imagery, and the blue segments correspond to the combination of both sets images [229]. This experiment focused on the identification of surface and

subsurface damage, but more importantly, the global visualization of both the color and infrared images as well as the identified damage information.

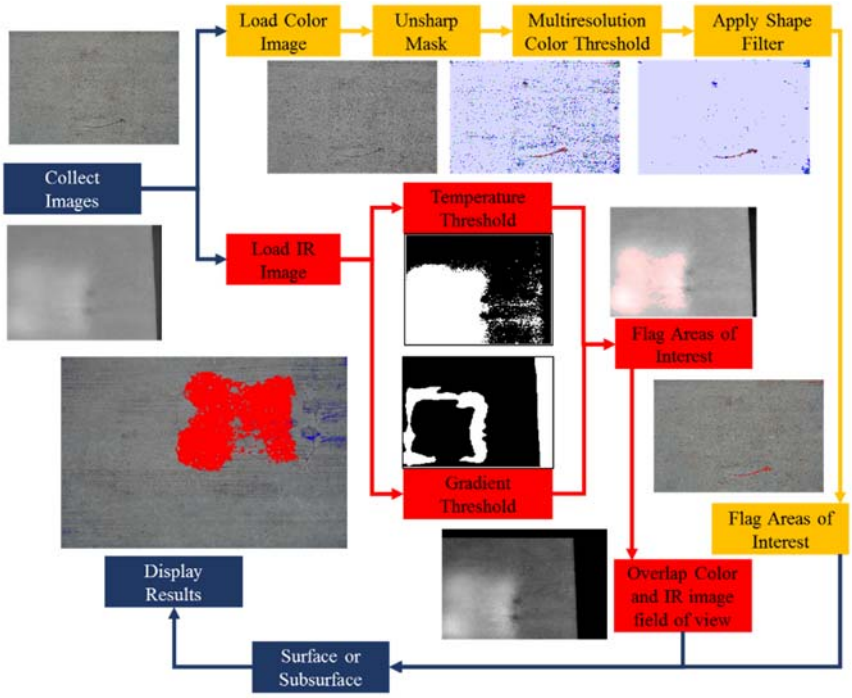


Figure 80. Data workflow of the damage identification algorithm for color and infrared imagery.

5.4.3 Results

The UAS images did not have the color and IR images synchronized perfectly which is easily visible in Figure 81 where the two images are overlaid on top of each other. The camera trigger signal was sent at the same time from the ground occurred, but the images are shifted. This means that the color and infrared cameras take different amounts of time to respond and capture the image after the 5V input sent from the flight controller to the camera systems.

Figure 81. Images taken during flight showing the time shift of image capture

This time shift in the image acquisition made the algorithm shown in Figure 80 inaccurate. The same process used in section 5.3.1 of manually aligning the images was not possible in this case because individual photos were taken, not video. This made it impossible to simply shift the images forward and backward to achieve an alignment. Therefore, the post-processing was done separately on the two datasets and the results could not be directly overlaid. Examples of each data set collected during the flight are shown in Figure 82 and Figure 83.

Figure 82. Color images captured during flight (top row) and post-processed images (bottom row)

In Figure 82, many of the false positives occur due to the change in material outside of the deck region. The shadow of the UAS causes false detection as well. The post processing algorithm does however make it easier to identify the cracks visually for a user who is tasked with looking through the pictures. Similarly in Figure 83, the IR images captured during flight improve the visibility of the hot areas. There were some false positives, however the processing still makes it easier for a user to identify potential delaminations.

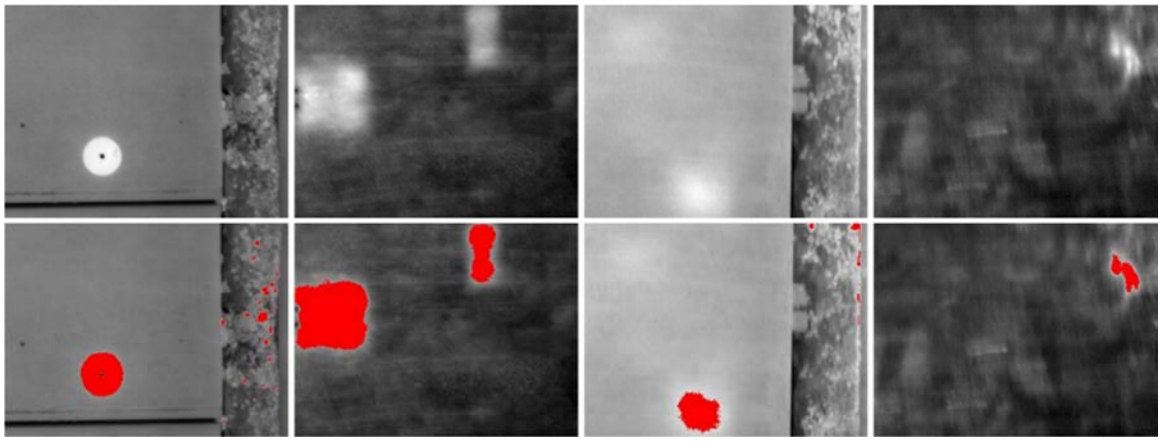


Figure 83. IR images captured during flight (top row) and post-processed images (bottom row)

Due to the misalignment of the color and infrared images, the mapping of the global damage was completed leveraging a rolling platform without assuming a known position of the camera. The images were analyzed with the same algorithms as the images above, with the added step of comparing the color and infrared images to distinguish between surface and subsurface damage identification. The top images in Figure 84 are the color images with the potential cracks are identified, the middles images are the infrared images with the potential delaminations identified, and the last row of Figure 84 shows the color

images with the potential delaminations highlighted in red and the potential cracks highlighted in blue.

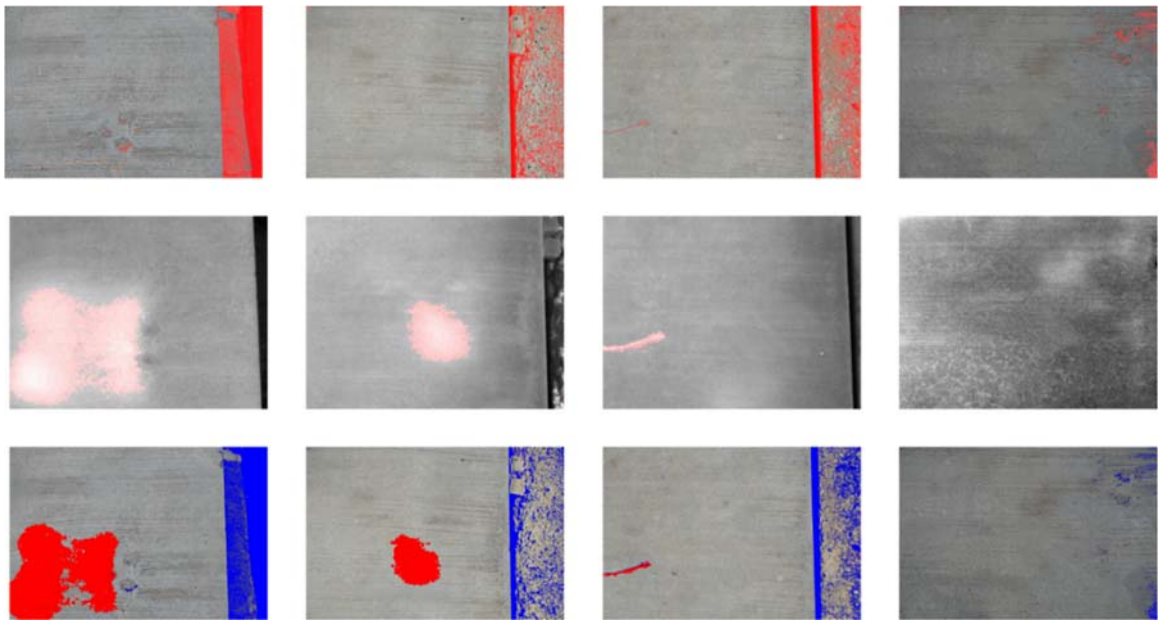


Figure 84. Identified damage from the ground based platform

From these results, it is clear that in many cases, cracks are identified by both the color and infrared damage identification algorithms. This leads to some misclassifications of the damage. In the future, better segmentation and machine learning algorithms could be applied to ensure this problem was kept to a minimum.

5.5 Global View of Local Data

5.5.1 Color and IR Without Radiometric IR Data

Figure 85 shows the images obtained after the stitching process was completed with Microsoft Image Composite Editor (ICE) using aerial images captured during flight.

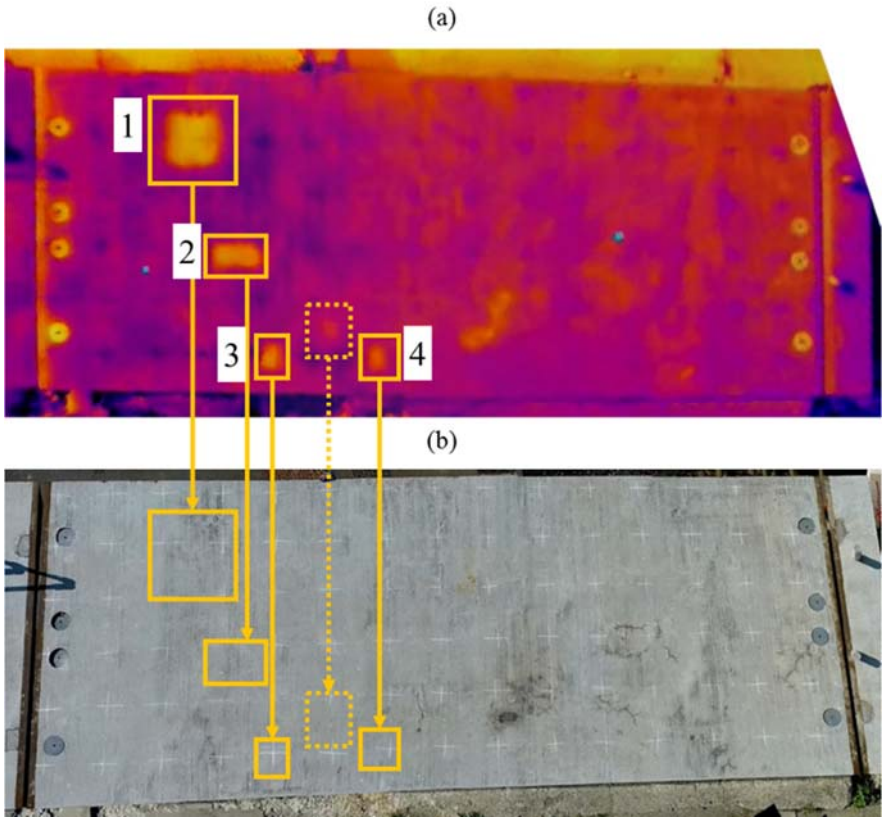


Figure 85. (a) UAS IR image mosaic output from Microsoft ICE. (b) UAS color image mosaic output from Microsoft ICE. The potential delaminations are highlighted in both images to confirm they are not surface damage.

It is easy to see that little to no parallax is present in the images. If the deck surface is not planar, problems with parallax could require a more advanced methodology to perform image stitching. These images were taken at higher altitude which made them easier to stitch, but for images taken closer to the surface, the program had more difficulty matching keypoints in the images. Therefore, a manual stitching code was written in MATLAB to stitch the images together and save the locations of each image and the manually selected points in the stitched representation. This resulted in a reduction in size of the global view that had the ability to recover the original high resolution image of an area of interest. The delamination locations in the stitched image match the locations of the

delaminations from the single image taken on the ground with the FLIR a325sc from Figure 75.

Image mosaics are used to obtain a global view over multiple images because local images are often not useful in decision making if they do not have a global context. In many scenarios in which a large area is covered by multiple images, it is difficult to visualize the location in which each image was taken from a global perspective, even if the images are being viewed in the order they were continuously captured. In order to address this issue, image mosaics can be used to provide information about the location of the individual images on the structure. There are several commercially available stitching software tools that can be used for creating panoramas; however, many of them rely on a constant camera location to stitch the images together [281]. In this special case, the projection onto the image plane goes through roughly the same focal point in all of the images (which represents a linear motion in cylindrical coordinates). When the camera's focal point moves between images, parallax becomes a problem because the background appears to shift in different images which causes automated feature detection methods to fail because the background can be correctly stitched instead of the foreground [282]. In this case, the camera position cannot be controlled in the same way. The bridge deck was planar with no visible background which allowed the same focal position assumption to be violated. If the deck was not planar, problems with parallax would occur and a more advanced method for image mosaicking would be required to account for keypoint matching issues. Microsoft Image Composite Editor (ICE) was used to stitch both the RGB and IR images together [225].

5.5.2 Color and IR With Radiometric Data

The images were then processed using VisualSFM [176] to create a 3D point cloud of the bridge deck from the color imagery obtained with the rolling platform. The bridge deck was assumed to be planar so a plane was fit to the points on the deck. The color images were used to identify key points in the images in which VisualSFM was used to calculate the three dimensional locations of the key points. These three dimensional points and the pixel coordinates of the key points in the images were used to calculate the transformation used to project the image onto the plane of the deck. Then the infrared images were mapped to the same field of view as the color images using the fixed distance and rotation between the two cameras. The same transformation used to project the color imagery onto the deck was used on the corresponding infrared images with the same field of view. The individual color and infrared images were then used to form an image mosaic and the damage information was displayed on the color for each image. Finally, the images with the identified damage were projected onto the plane of the deck and displayed in the same way as the color and infrared mosaics. The algorithm is heavily dependent on the color and infrared images to be captured simultaneously or in the same location. The distance and rotation of the color with respect to the infrared camera also could not change during the test. Since structure from motion does not rely on the position of the camera or the way the camera is moved as explained in section 2.4.6, it is possible to obtain the images from this moving platform and can in the future extend to UAS imagery with synchronized image capture.

Figure 86 (a) and (b) show the color and infrared image mosaics respectively, with the damage manually identified in red for delaminations and blue for surface damage.

Figure 86 (c) shows the automatically identified damage information obtained using the algorithm in Figure 80 with red indicating subsurface damage and blue indicating surface damage. The manually identified damage in boxes is also overlaid onto the automatic identification of the damage.

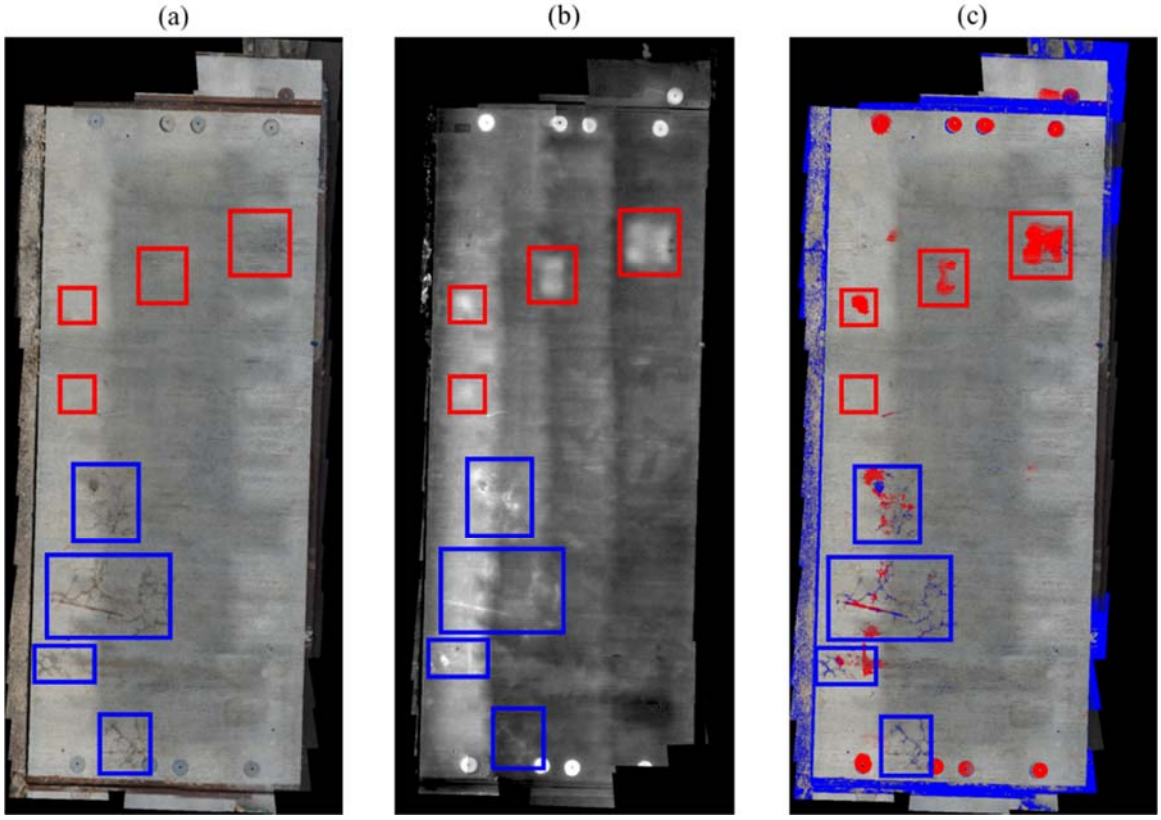


Figure 86. (a) Color stitched image. (b) Infrared stitched image. (c) Color stitched image with automated damage identification displayed in red and blue.

The global view of the structure showing the automatically highlighted damage information is important to assess the condition of the bridge deck. The size, location, and type of damage must be reported to the bridge maintenance crews or inspection teams in order to fix the right area of the structure at the right time. The use of fast data acquisition

platforms and automated algorithms for identification make the use of SHM for bridges in the way we see it used in other industries possible in the future.

5.6 Field Validation

The field test explained in section 4.5 was used to also test a delamination identification algorithm. Rather than use the narrow field of view ICI infrared camera on foot, a rolling platform shown in Figure 87 was constructed to speed up the data collection process.

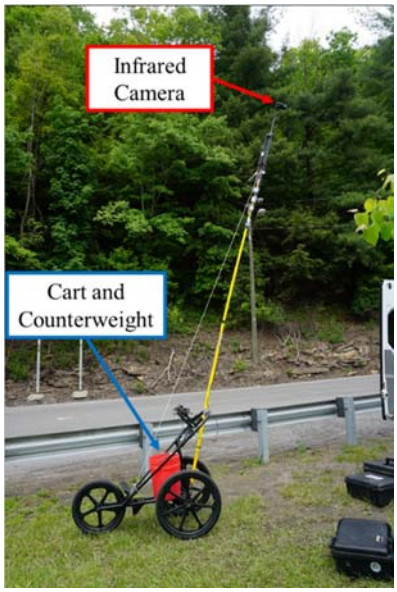


Figure 87. Setup for collecting infrared data

The camera had to be placed very high to efficiently collect data and the structure could not support two cameras for safety reasons. Therefore, the technique in the previous section could not be completed for a global infrared map because the color and infrared images were not taken at the same time or location. This limited the results to the analysis of local data, but the color imagery could be matched to some of the infrared imagery given specific features in the deck that could be identified manually. The examples in Figure 88

and Figure 89 show where on the deck the color and infrared image are located using the global view in shown in 4.5.

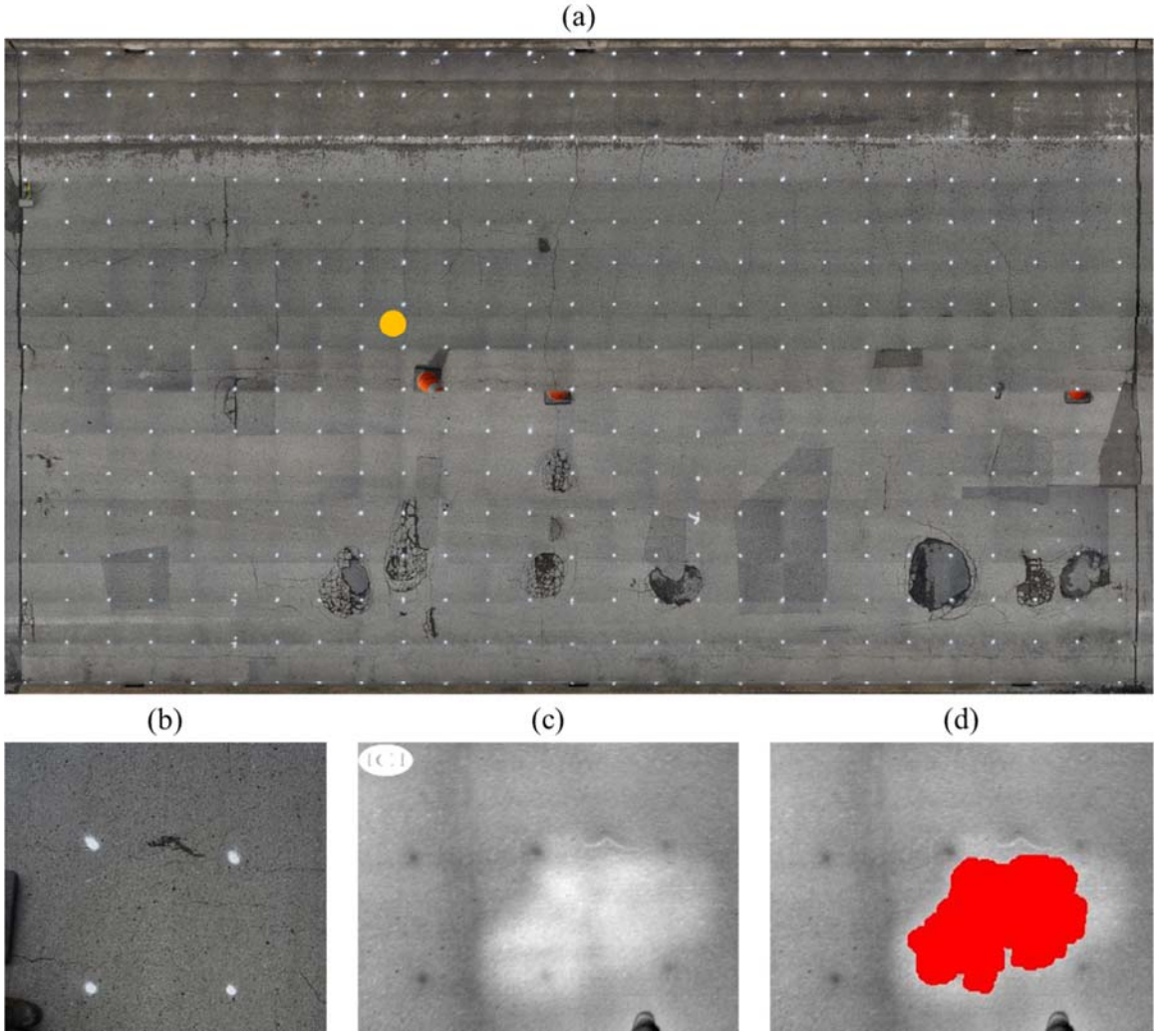


Figure 88. Image mosaic of the bridge deck with location of interest. (b) Color image at the highlighted location. (c) Infrared image at the highlighted location. (d) Potential subsurface damage identified by the algorithm.

The images shown in Figure 88 and Figure 89 are the global view of the deck (a), the original color (b) and infrared images (c) taken at each point displayed in (a), and the

identified potential delamination. The delaminations were identified using infrared part of the algorithm described in Figure 80.

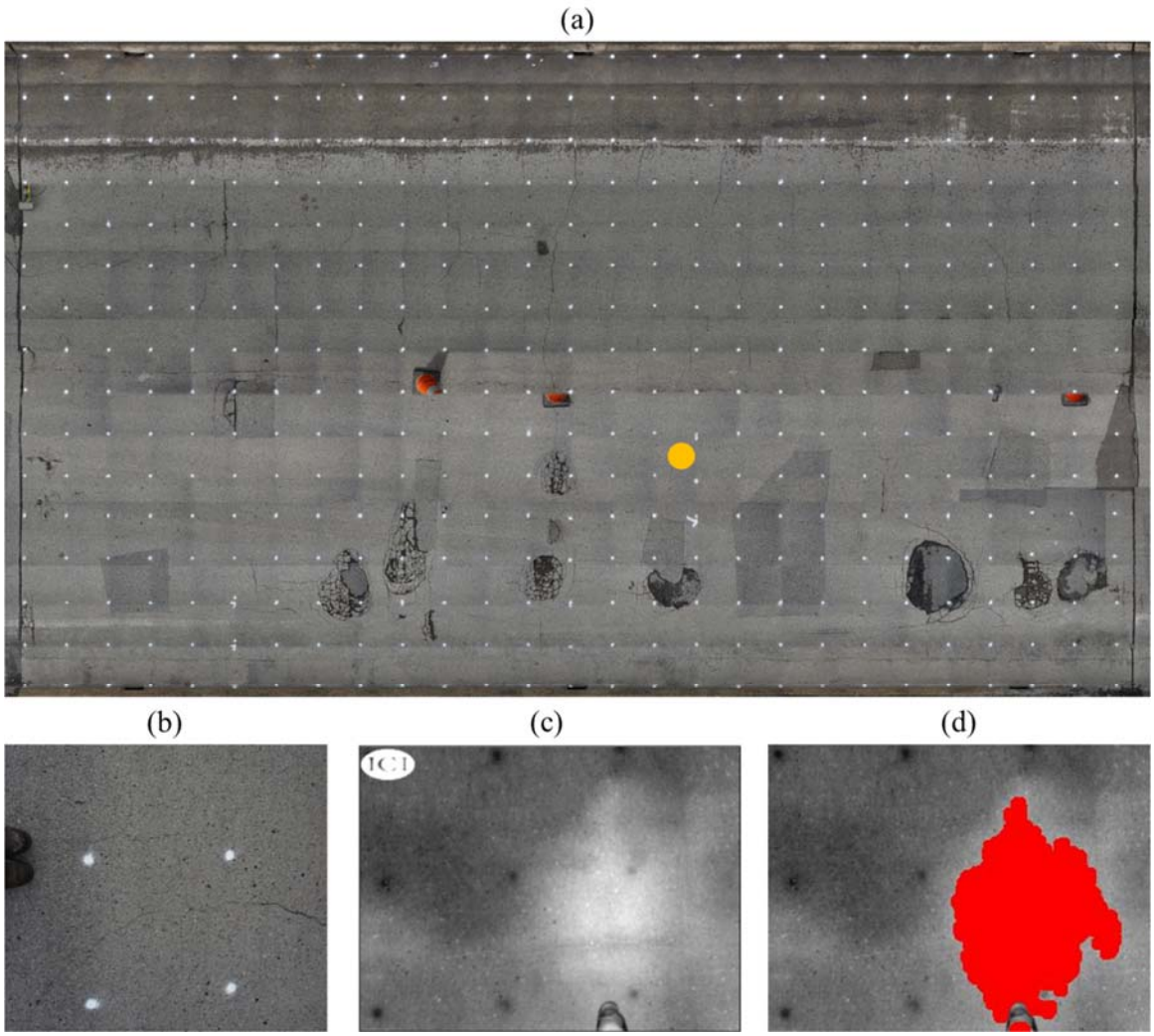


Figure 89. (a) Image mosaic of the bridge deck with location of interest. (b) Color image at the highlighted location. (c) Infrared image at the highlighted location. (d) Potential subsurface damage identified by the algorithm.

Chapter 6 Deformation Measurements

6.1 Introduction

There are several types of contact and noncontact sensors that are used in the field for displacement and strain measurements. Among these are string potentiometers, lasers, accelerometers, optical methods, and GPS [163, 164, 283, 284]. Though contact sensors can be more accurate and be more reliable, they are often difficult to install, difficult to interpret, and only provide information at a few points along the structure. The use of a mobile platform with noncontact measurement systems to obtain more information is highly desired in cases such as bridge SHM Though visual inspection has been reliable, funding limitations, the slow speed, and lack of repeatability between inspectors makes alternative methods of inspection seem attractive [27, 251]. In this context, different types of UAS based remote sensing have been applied to measure structures in a laboratory setting to measure the displacement of a structure using targets and speckle patterns [174].

6.2 Displacement Gauge Calibration

In each of the experiments described, string potentiometer displacement gauges were used as the ground truth for the image based measurements. The gauges were calibrated prior to each test to ensure the correct multiplication factor was used to determine displacement from the voltage. This ensured that the calibration constant was accurate in case the calibration constant changed. Generally this is not an issue, but the gauges were often used for other experiments in the field and when they get bumped and experience temperature changes, it is possible that the constant will change very slightly. This was completed using the calibration jig shown below in Figure 90 that was used to find the

calibration constant. The red box shows the voltage output to the computer, the orange box highlights the displacement gauge, the attachment connecting the jig to the displacement gauge is highlighted in green, and the displacement control highlighted in blue controls the exact displacement to 0.02mm accuracy. There is a linear relationship between voltage and displacement so a line was fit to the voltage outputs at different distances. The slope of the line was set to be the new calibration constant associated with that displacement gauge.

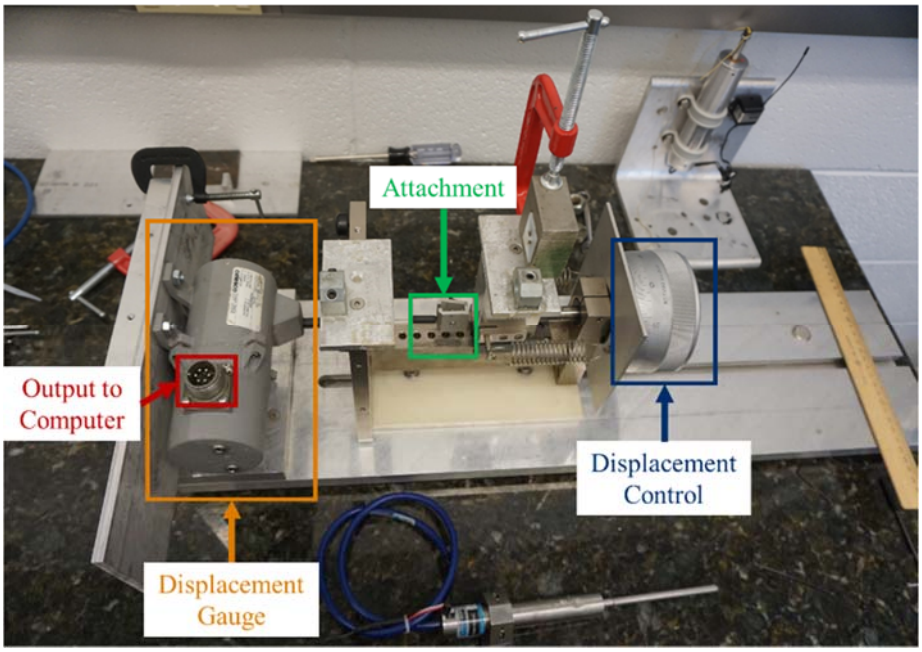


Figure 90. Displacement gauge calibration jig

The calibrations were completed with the displacement gauge roughly 2 inches from the fully retracted position because the gauge should be linear through the range of 1 to 5 inches. The calibration ranged from 0 to 25mm in 5mm steps from the starting point. An example of the calibration data coming from the displacement gauge is shown in Figure 91.

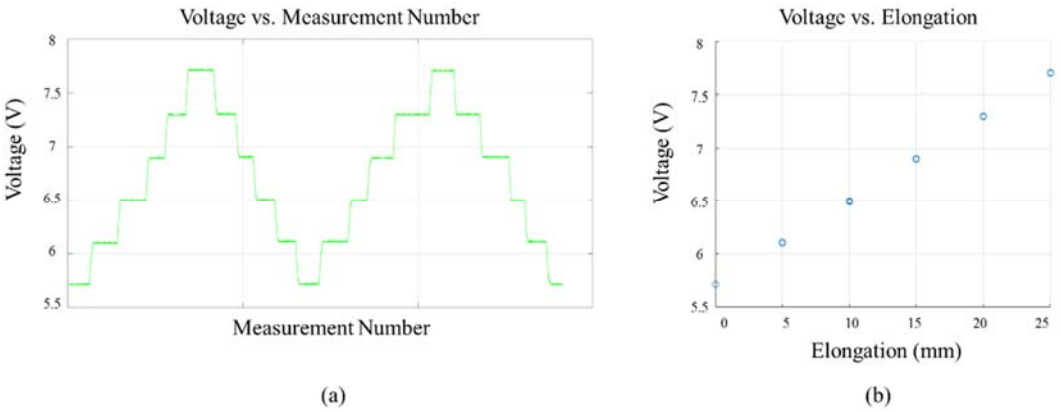


Figure 91. (a) Default voltage values out of the displacement gauge. (b) Curve used to determine the calibration constant.

The default values out of the gauge shown in Figure 91 (a) were used to create the curve shown in Figure 91 (b). A line was fit to the points in Figure 91 (b) and the slope was used as the calibration constant to determine the deformation at a given voltage.

6.3 Pointwise Deformation Measurements

6.3.1 Equipment

The commercially available Parrot AR 2.0 shown in Figure 92 was used to complete the preliminary experiments. This small quadrotor was equipped with one camera used for stability and position hold using optical flow during the flight of the UAS and a second camera facing forward used to obtain imagery. The UAS was made of carbon fiber, nylon, and foam, and used four brushless motors with a total vehicle weight of 420 grams. It was powered by a 3S battery which provided about 8 to 12 minutes of flying time in the laboratory setting. Furthermore, the UAS could be controlled by any Wi-Fi device with the application AR FreeFlight. In addition to the optical flow camera, the system's flight control consists of a three-axis gyroscope, an accelerometer, a magnetometer, a pressure

sensor, and an ultrasonic sensor with a resonance frequency of 40kHz to measure UAS altitude up to 6 meters [285].

Figure 92. Parrot AR 2.0 UAS [286]

The forward facing camera fixed to the front of the UAS has a resolution of 1280x720 pixels, a 92º wide angle lens, and recorded video at 30 frames per second (fps) while taking still images simultaneously. The pictures and videos were transmitted to the device controlling the UAS or saved on an onboard flash drive. The bottom facing camera of the UAS, used for control and collecting data, had a resolution of 640x320 pixels and captured images at 60 fps. The pilot could switch between the camera views from the ground using the app during flight. The ultrasonic sensor and the camera fixed at the bottom of the UAS assisted with flight stability, but it should be noted that the user could program the UAS to maintain altitude and position using both the ultrasonic sensor and downward-facing camera [286].

The deformation measurements obtained from the data collected by the UAS were compared and validated with a commercial optical metrology system (TRITOP, distributed by Trilion Quality Systems) that provided up to 50 microstrain accuracy. This optical

metrology system utilized a single camera to track user-placed fiducial markers, that could vary in size, on a given field of view by using a photogrammetry framework, presented in section 2.4.5, which assumes no marker movement between stages [287]. Specifically, the system identifies ellipses, due to oblique views of the circular markers depending on the sensor location, and then calculates the centroid of each marker leveraging the perimeter of the ellipses. Defined in this way, a set of digital points was used for deformation measurements with the scale provided in the field of view by both appropriately defined scale bars as well as an additional set of coded or reference markers, as referred to by the manufacturer. The perimeter of each marker was determined with the grayscale image and only markers with a minimum diameter of ten pixels were used in the calculations. In general, larger diameters in pixels provide more accurate centroid locations. This approach yields relatively low-density point clouds when compared to full field DIC or structure from motion, but it provides the accuracy associated with DIC with the flexibility of structure from motion. The calculation of the 3D coordinates and deformation using multiple camera positions for a given target in the field of view require that no changes occur within the scene while the images within a stage. The TRITOP system leveraged a 16 megapixel digital single lens reflex (DSLR) camera which was validated with a corresponding full-field 3D DIC system leverging two 5 megapixel stereo camears and the ARAMIS DIC software [288]. The photogrammetry approach does not require precise control of the camera position by leveraging many camera views, which is promising for application on a UAS.

Since it is possible to carry LiDAR on UAS platforms, the UAS image based measurements were compared with a 3 dimensional measurement from the X-BOX Kinect,

which was capable of providing RGB images with distance measurement associated with each pixel. The Kinect used an infrared laser projector to generate depth information for each RGB image pixel. The Kinect color camera had a resolution of 480x640 pixels which was used to determine the in plane coordinates leveraging the depth measurements. It measures data relative to its own coordinate system, which in conjunction with the known position of the Kinect, was used to obtain information on a globally defined coordinate system. The Kinect sensor has a minimum operating distance of 0.5 m from the target and two millimeter accuracy at one meter distance which degrades with increasing distance up to seven centimeters at a distance of five meters [289]. The Kinect was used in the place of a LiDAR system due to its low cost and data rich output. Though it cannot be used outside, there are commercial systems that can be implemented on a UAS with higher payload, such as the Velodyne VPL-16 [290]. Since the UAS was incapable of carrying the camera used for the TRITOP software and the Kinect due to the weight of the systems, the two systems were tested separate from the UAS for the proof of concept.

6.3.2 Data Post-Processing

The images captured with the UAS camera were severely distorted because of its wide angle lens, as displayed in Figure 93 (a). The distance variability per pixel throughout the image for a fixed distance perpendicular to the target. Therefore, raw captured images were post-processed by using a calibration algorithm explained in section 2.4.3. The resulting corrected image is shown in Figure 93 (b). This method ensured that the distance per pixel throughout the image taken perpendicular to the camera is not affected by the lens distortion, making it possible to place a scale bar in that plane within the field of view of the image instead of the checkerboard.

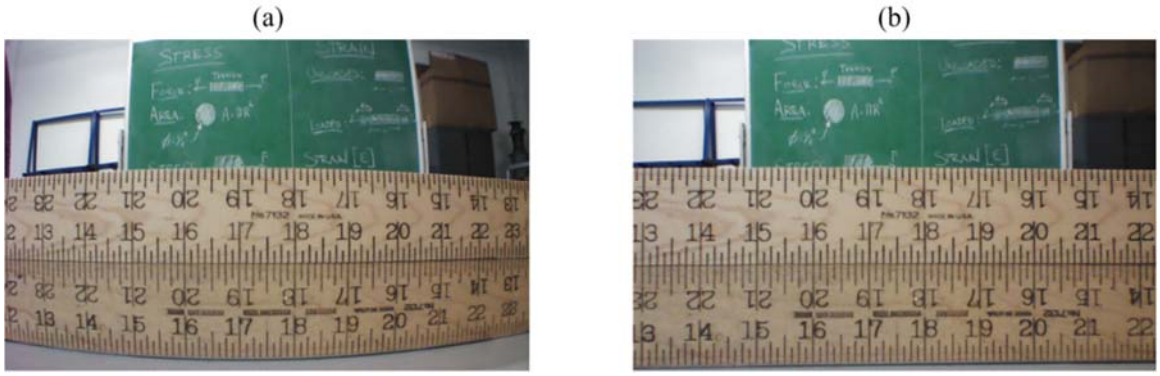


Figure 93. a. Image obtained with the built-in UAS camera showing issues associated with measurement in images with lens distortion present. b. Corrected image after camera calibration

Prior to experimentation on a structure, flight testing was performed to determine the effect of hovering on the image clarity. An example shown in Figure 94 is an image taken while the drone was hovering with the scale bar and a set of uncoded, orange.

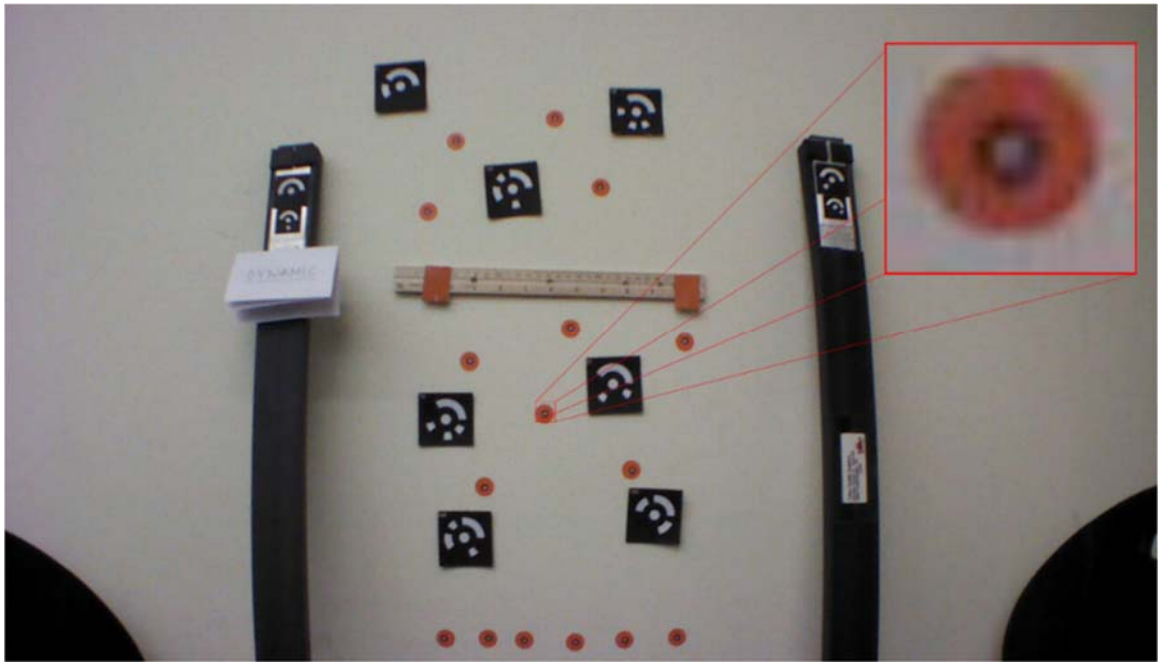


Figure 94. Image obtained while the drone was hovering

The TRITOP coded markers and scale bars were also in the field of view of the image. The TRITOP black and white circular markers were placed in the center of each of the orange fiducial markers. The marker highlighted in Figure 94 shows the marker magnified to 8 times the actual size in the image to determine the amount of blur associated with the UAS vibration. In this case, the wide angle lens and blur associated with the image could be neglected. However, studies have been completed to help correct these problems with image blur using UAS [291].

A computational algorithm was developed in MATLAB to compute the coordinates of the fiducial markers in images captured by the UAS. Since there was not a forward facing ultrasonic sensor to determine distance and angle of the UAS with respect to the target, the position of the camera with respect to a given target presented a practical challenge for this approach. This challenge was overcome assuming all of the images captured by the UAS were taken perpendicularly to the target. Therefore, the aircraft was carefully positioned during flight so only pictures taken perpendicularly to the target would be used for measurement purposes. It was assumed that images taken within 5 degrees from perpendicular to the target were acceptable for measurement. However, it is also known that the error increases as the markers get further from the center of the image. Since the angle was impossible to measure during flight with the equipment used for these experiments, the estimation of the UAS perpendicularity to the wall was made from multiple pilot and viewer positions. The perpendicularity assumption made it possible to eliminate the coded markers from the scene, which would be needed in the case that oblique angles of observation were required. In a system with higher payload, a high resolution

DSLR camera could be used to identify coded markers which would then allow for the processing of images taken at different angles. The coded markers would be required for matching purposes.

The algorithm steps to obtain measurements from the UAS images are shown in Figure 95. The scale bar was placed in the image with rectangular markers on either end for easy identification and the scale bar was assumed not to move throughout the test. The distance between the scale bar markers was a parameter set prior to testing. Two images were imported where one was assumed to be the reference in which all of the other images were compared to and the second image was used for comparison. The images were subsequently corrected for lens distortion and markers were identified using their color. The markers were chosen to be orange because it is one of the easiest colors to identify in an image and non-orange areas of the image were eliminated using the color, size, and shape of the markers. The image was converted to a binary image using a color threshold and the centroids were calculated and tracked with the algorithm. The size of the markers in pixels within the image varied with the distance from the camera to the markers. The minimum acceptable marker radius was 10 pixels for each test regardless of the physical dimensions of the marker. Greater numbers of pixels encompassed by the marker resulted in more accurate centroid locations that would be used for the measurements. The Hough transform was used to identify the shape of the markers [128], which determined which markers corresponded to the scale bar and provided measurement points with a known scale within the image. The relative position of each marker was measured in a local coordinate system defined by the scale bars. This aligned the images to a common FOV to quantify the deformation.

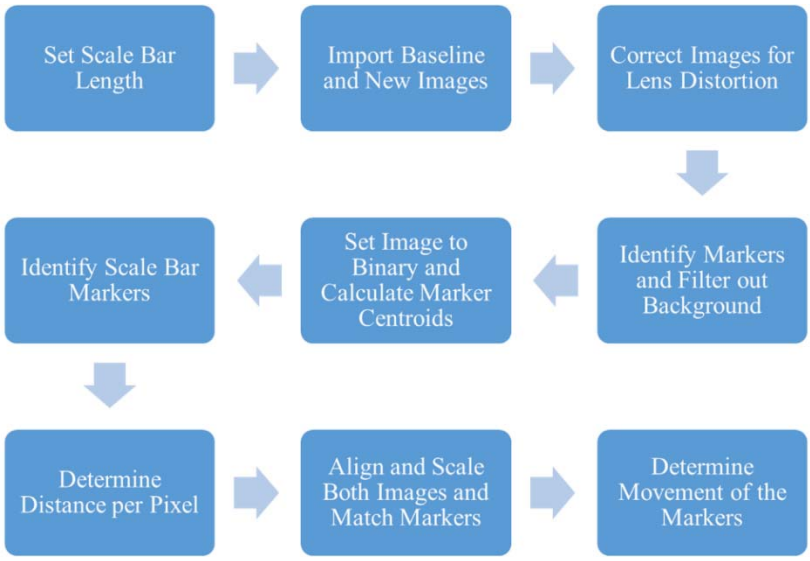


Figure 95. UAS MATLAB algorithm steps to image measurements

The X-Box Kinect captured 3D coordinates at every pixel in the image and leveraging the additional information, another algorithm was developed in MATLAB. Each marker centroid was calculated from its area as in the previous algorithm. The distance measurements provided by the Kinect were used to determine the horizontal and vertical coordinates of each marker instead of using the scale bar. The distance measurements from the Kinect would output zero readings due to range limitations. Therefore, a local average of the distances around each centroid was used to estimate the actual distance to the marker for all non-zero distance readings.

The algorithm steps to obtain measurements from the Kinect images are shown in Figure 96. The baseline Kinect data and the data used for comparison were first imported into MATLAB. The Kinect data was a color (RGB) image and distance measurements to each pixel in a fourth channel. The markers were identified in the images by their color and the image was filtered and converted to binary using the color based thresholds. Then, each

of the marker centroids were calculated in pixels with the origin located at the center of the image. The angle θ was calculated counterclockwise from the positive x axis for each marker. Distance measurements from the Kinect to each marker were determined using only the non-zero measurements for all pixels belonging to a marker. Equations (6.1) and (6.2) were used to calculate the x and y distance in millimeters from the Kinect using the distance measurements and the locations of the centroids in pixels. The matching of the markers was accomplished using Euclidean distance assuming that the movement of each marker was small enough that there would be no tracking errors. Finally, the movement of each marker was calculated.

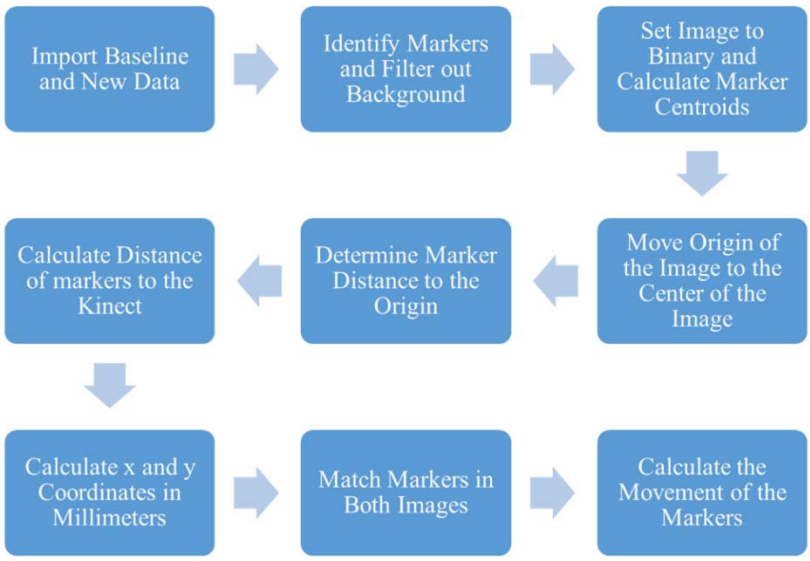


Figure 96. Steps to the Kinect measurement MATLAB algorithm

The distance measurements and the known, fixed position of the Kinect allowed the measurements to be obtained without the use of a scale bar. Similar to LiDAR, UAS localization must improve significantly for this approach to be implemented in the field for

inspection applications. The major advantage of this approach is that global measurements are obtained making it feasible to determine if the entire structure has moved in a way that cannot be determined with local measurements. Examples of this include settlement, deformation caused by scouring, or shifting of the roadway or pier [96, 292]. In this experiment, the sensor did not move, so the local and global coordinate system were the same. In mobile applications, the exact position of the sensor must be well known for deformation measurements to be feasible. Systems with real-time kinematic global positioning systems (RTK GPS) can achieve centimeter accuracy positioning information with respect to the reference which has been used for geometric mapping of structures, but higher accuracy may be required to measure movements of the structure [36].

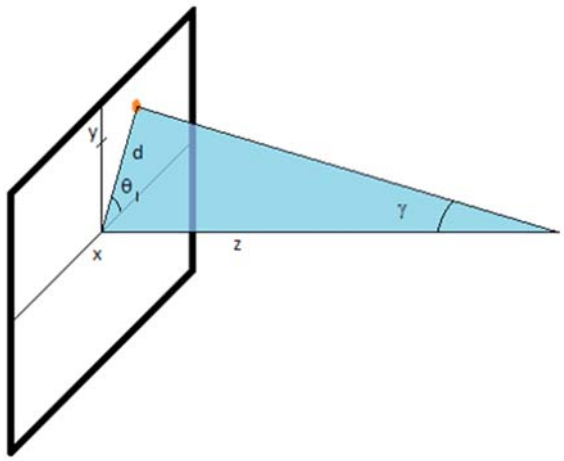


Figure 97. Diagram of the distance measurements used in image post processing with the Kinect sensor

The diagram in Figure 97 provides a visual representation of equations (6.1) and (6.2) which were used to calculate the horizontal and vertical position of the markers in a local coordinate system centered at the Kinect sensor. The angle θ was determined from

equation (6.3), the angle γ was determined from equation (6.4), and the linear distance in pixels (d) from the center of the image to each marker was calculated with equation (6.5). After determining the x, y, and z coordinates in millimeters, deformation was calculated between the sets of images.

$$x_m = \tan(\gamma) \cdot z_m \cdot \cos(\theta) \tag{6.1}$$

$$y_m = \tan(\gamma) \cdot z_m \cdot \sin(\theta) \tag{6.2}$$

$$\theta = \arctan\left(\frac{y_p}{x_p}\right) \tag{6.3}$$

$$\gamma = d \cdot \theta_p \tag{6.4}$$

$$d = \sqrt{x_p^{\,2} + y_p^{\,2}} \tag{6.5}$$

The angle per pixel for the Kinect, θ_p, was assumed equal to 0.089^o which was computed by dividing the experimentally determined horizontal and vertical field of view angles by the number of pixels in each direction for both RGB and IR data. The variables x_p and y_p are the marker centroid coordinates in pixels and variables x_m, y_m, and z_m are the local coordinates of the markers in millimeters.

6.3.3 Perpendicular Camera Angle

Deformation measurements were performed both statically and while the UAS was hovering indoors. The first setup using a cantilever aluminum L-beam is show in Figure 98. The images were taken statically with both the UAS camera and the Kinect sensor. The limited space in the laboratory prevented the UAS pilot from capturing images while the system was hovering. The purpose of the test was to validate the performance of both the UAS and Kinect algorithms. TRITOP was used as the ground truth measurement for this

setup. The orange markers were used by the developed algorithms and the black and white TRITOP markers were placed at the center of each orange marker for direct comparison.

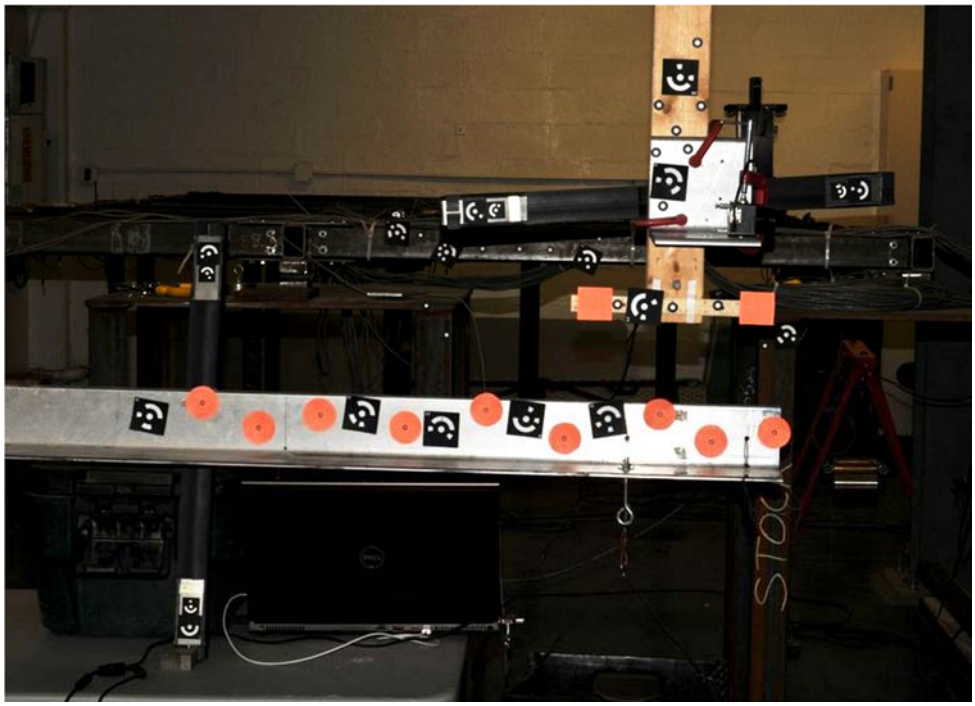


Figure 98. An aluminum cantilever beam used for static measurements with vertical loading imposed near the tip of the beam.

Figure 99 shows a comparison between TRITOP measurements and the measurements obtained from the developed algorithms for both UAS and Kinect images. The solid black lines in the figure show the measurements obtained from TRITOP which appear to be linear because the images were only taken at the free end of the beam. The error bars were computed by taking three measurements with no load after loading and unloading. The origin location was set to be at the centroid of the left orange scale bar marker. It was determined from the results that the Kinect had an accuracy of ±3.5mm, and the UAS had an accuracy of ±2mm.

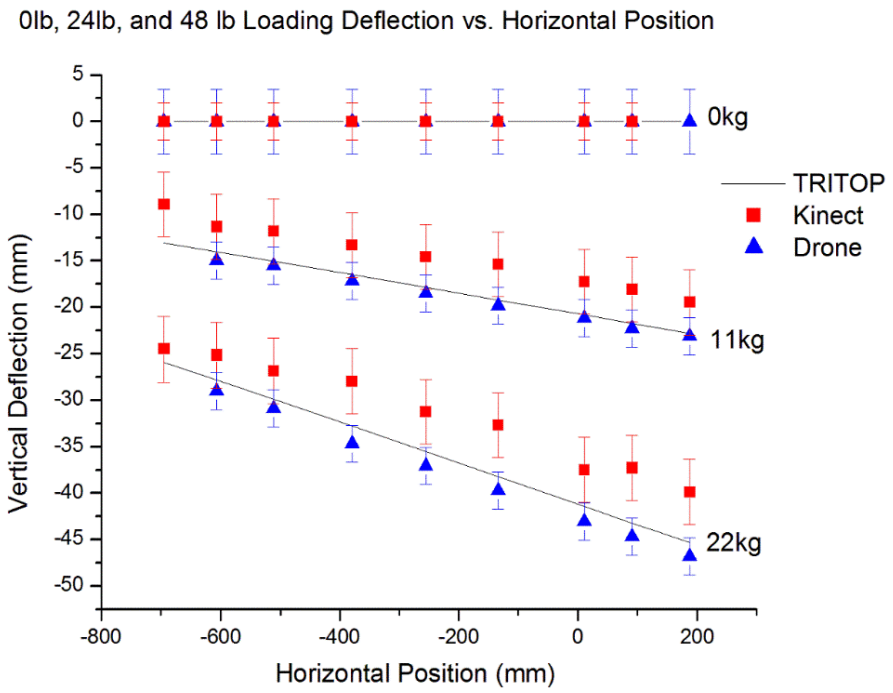


Figure 99. Deformation results for unloaded (0kg load baseline) and loaded with 11kg and 22kg aluminum L-beam

The inaccuracy of the Kinect increased as the deflections increased, as shown for the measurements obtained for both 11kgs and 22 kgs of loading. These inaccuracies could be explained by the angle per pixel, explained by Figure 97, which was assumed to be 0.089°, being inaccurate or not valid in practice. The offset between the RGB and IR sensors was not taken into account for these experiments and could lead to inaccuracies in the assumed value. However the offset between the sensors is 25mm in the x direction, so it was not expected to affect the vertical component of the measurements. Since the vertical component was the focus of the experiment, this source of error should have a limited effect on the results due to the offset. Another source of error for the Kinect is lens distortion in the infrared pattern. Since the infrared pattern projected by the Kinect passes through a lens, there is distortion associated with it. The infrared lens distortion was neglected for

this experiment because a checkerboard used for calibration is not visible. Many commercial LiDAR systems scan differently and often do not require a lens to collect the data which would eliminate this problem. The deformation measurements obtained from the UAS imagery consistently followed the deflection profile, while the observed error range suggests that a camera with higher resolution could allow for higher accuracies assuming the proper sensor and optics are used.

Similarly, deformation measurements were performed on a laboratory scaled steel grid [42], shown in Figure 100. In this case, UAS images were captured both statically and during flight while hovering. The deck was loaded and unloaded three times, and three measurements were for each stage with the UAS in a static and hovering configuration and also with the X-Box Kinect. One ground truth measurement was taken with TRITOP as a baseline to compare the image based measurements obtained from the Kinect and UAS static and hovering.

Figure 100. (a) UAS hovering near the steel deck. (b) Picture captured by the UAS camera

The results of the test are shown in Figure 101. Nine measurements from each of the same load of 102kg were compared using a t-test, and no statistically significant

differences between the means of each measurement system were observed. The error bars were determined using a student t-distribution with a 95% confidence interval and eight degrees of freedom using each of the nine measurement points (orange dots) from each loading. Since the measurements were expected to be different for each of the markers, the markers were not compared to each other.

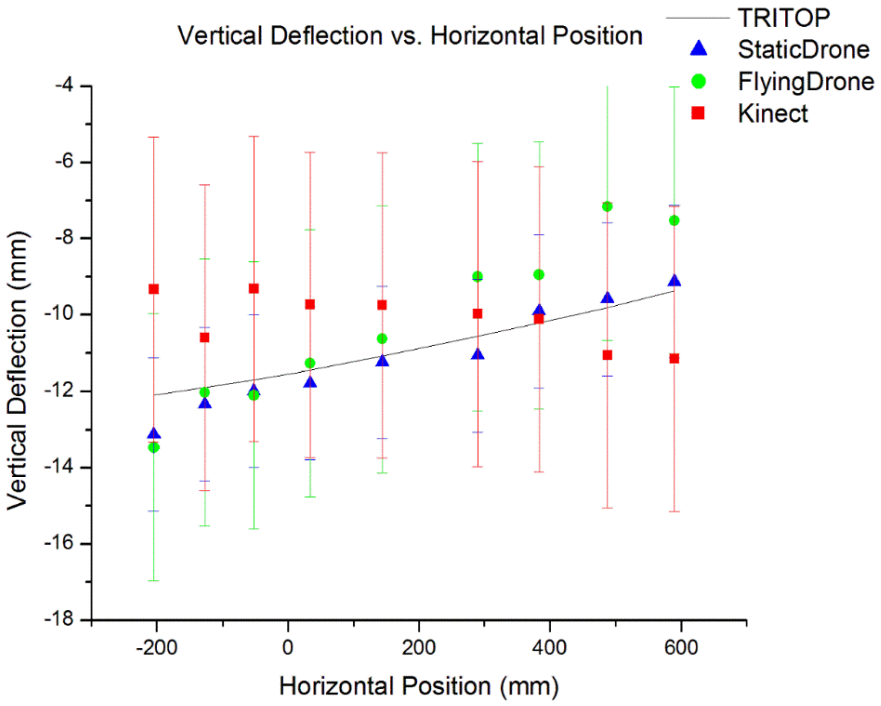


Figure 101. Deformation results for loaded steel deck from both the UAS static and hovering measurements and the Kinect measurements

As it was in the aluminum L-Beam test, the zero horizontal position was located at the left scale bar marker. Based on these measurements, it was determined that the X-Box Kinect had an accuracy of ±5mm, the UAS in a static configuration had an accuracy of ±2mm, and when hovering, the UAS had an accuracy of ±3.5mm.

The measurements obtained with the Kinect seem to have an opposite trend when compared to the expected deformation; however, due to the limited resolution of the system, error bars suggest that the structure did not move enough for the Kinect to reliably detect the movement. The error was approximated to be ±5mm for this testing configuration and the measurements obtained from the Kinect sensor fluctuated between -9 and -12 which does capture the overall movement of the setup. The aluminum L-beam test provided a better representation of the capabilities of this system because the deformations were detectable with the sensor. The system could have been placed closer to the steel grid to increase the resolution to obtain better measurements if the Kinect could obtain measurements at working distances less than one meter from the structure. However, when the Kinect is placed closer than one meter to an object, the distance measurement to that object read zero.

Since the UAS was manually piloted, the system may not have been perfectly parallel to the setup when the image was captured during flight, which added a significant amount of error compared to the static setup. This would be an even bigger problem in the presence of wind, but a UAS with higher payload could carry a high resolution camera and use photogrammetry to its full potential with both coded markers and uncoded markers making angles to the target possible. Furthermore, a vibration dampening system and gimbal could be used to stabilize the images during flight which would decrease the amount of blurry images captured during flight.

6.3.4 Field Demonstration

A field experiment was performed on a pedestrian bridge to demonstrate the potential of the UAS to fly around an actual structure and obtain aerial imagery that could be

leveraged for damage detection and analyzed by the algorithms to obtain deformation measurements.

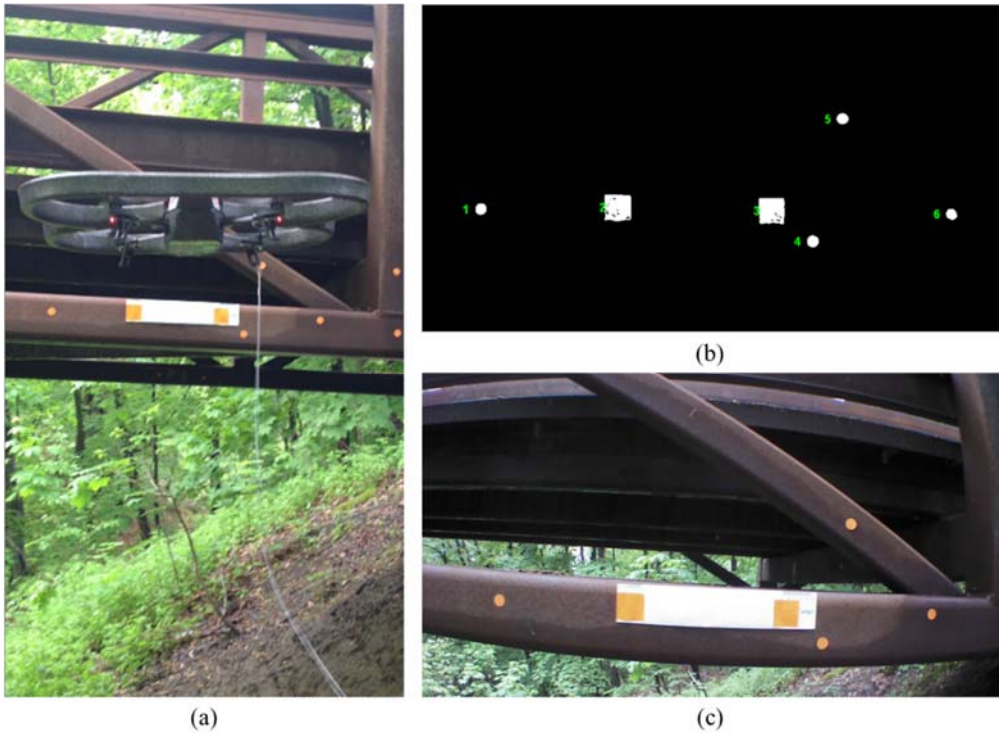


Figure 102. (a) Picture of the UAV hovering near the pedestrian bridge; (b) Image taken by the UAV showing the markers previously placed. (c) Markers identified using the image processing algorithm

Specifically, Figure 102 (a) shows the UAS hovering alongside the bridge, and Figure 102 (b) shows an image captured during the flight of the orange markers placed on the structure prior to the flight. These images were imported into the UAS algorithm and the markers were successfully identified as shown in Figure 102 (c). Marker identification is one of the most important components of the measurement algorithm because partially identified markers or incorrectly classified shape result in severely incorrect measurements. This demonstration was only used to ensure that the algorithm could identify the markers

correctly on a structure outside of the laboratory. No loads were applied to the pedestrian bridge so deformation measurements could not be calculated.

6.3.5 Oblique Camera Angle

Since a perpendicular angle to the structure is not always feasible, a method was explored for obtaining measurements at large angles. Using the same laboratory steel grid as in section 6.3.3, static deformation measurements of the deck were performed using images captured on a tripod. Prior to the experiment, calculations were performed to determine the depth of field and expected blur associated with each of the markers. This was done to ensure that the markers were in focus enough to be correctly identified. The length of the deck was just under 6.5 meters and therefore the depth of field was required to be at least 6.5 meters to accurately determine the location of all of the markers in the image. The depth of field is defined by the nearest and farthest distance from the target that an object is still in focus with the same camera settings. These distances can be determined using the circle of confusion (c), focal length (f), F-number (N), and object distance (z). The circle of confusion was based on the physical size of the sensor and the number of pixels. In this case, the physical size of a pixel on the image sensor was set to be the circle of confusion because any amount of blur within a pixel cannot be detected. The hyperfocal length (H) is the focus distance where all of the objects further than that distance appear to be in focus. This value was calculated using equation (6.6) and is necessary to determine the depth of field [293].

$$H = \frac{f^2}{N \cdot c} + f \tag{6.6}$$

The closest (D_N) distance in focus was calculated using equation (6.7) and farthest (D_F) in focus was calculated using equation (6.8) [293].

$$D_N = \frac{z \cdot (H - f)}{H + z - 2 \cdot f} \tag{6.7}$$

$$D_F = \frac{z \cdot (H - f)}{H - z} \tag{6.8}$$

The steps to obtain the measurement are below in Figure 103. The first step was to calibrate the camera to find the camera intrinsic matrix and correct the images for lens distortion. The markers were then identified using multiresolution edge detection and shape filtering, followed by circular Hough transform. The markers were then matched using the pattern and spacing of the dots on each marker. The three dimensional coordinates were obtained through triangulation limiting the movement to the directions the marker was expected to move. The camera coordinates were then mapped to the same coordinate system based on the known locations of the fixed markers.

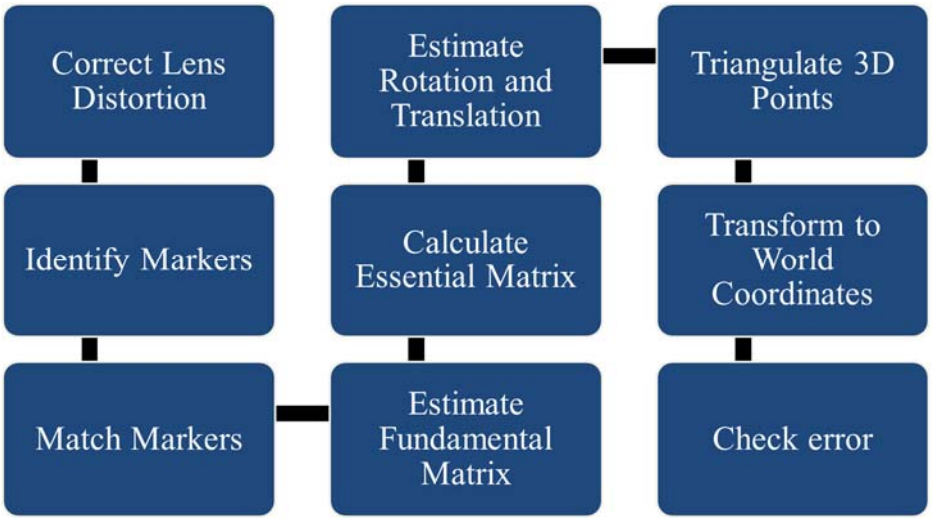


Figure 103. Steps to obtaining measurements from images taken at an oblique angle.

A preliminary experiment was completed to validate the approach. An image captured during the experiment is shown in Figure 104. The two markers in the red boxes

were fixed during the experiment. The center marker in the green box was moved 5mm, 10mm, and 20mm.

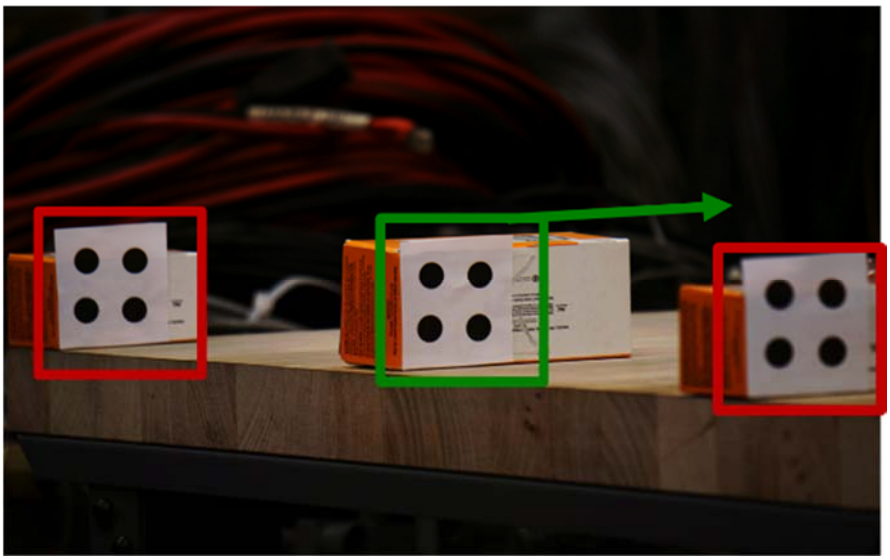


Figure 104. Small scale point measurement validation.

The camera parameters for this experiment were a 210mm focal length, aperture of f/6.3, ISO-100, and shutter speed of 1/30 seconds. After the experiment the images were processed to determine the movement using the steps outlined in Figure 103. The metric used to determine if the triangulation and transformation to world coordinates was acceptable was a movement of less than 0.25mm for all 8 dots on the fixed markers. After applying this criteria, 65 comparisons for 0mm movement, 34 for 5mm movement, 15 for 10mm movement, and 37 for 20mm movement were determined to be good measurements. The results of the experiment are shown in Table 9 and Figure 105. The colors correspond to each movement. The dots are the image measurements and the x's are the actual movement.

Table 9. Results of small scale validation experiment.

X Location (mm)	Mean Camera Marker Movements (mm)		
	Marker 1	Marker 2	Marker 3
0	0.00	-0.21	0.00
5	0.00	5.07	-0.00
10	0.00	10.47	-0.00
20	0.00	20.07	-0.00

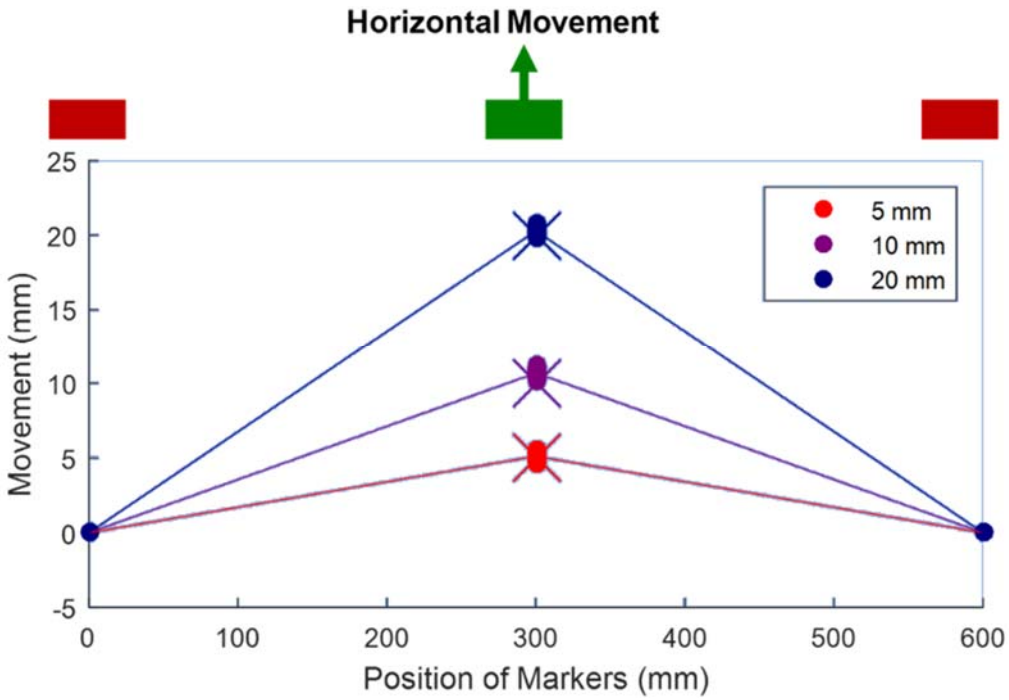


Figure 105. Results of the validation experiment.

Since the movement was done by hand and measured with a ruler, it is possible that some of the errors were a result of poor movement control. In general, the measurements performed very well and the grouping was very tight across all of the images that produced an acceptable reconstruction.

After validating the method, the larger scale experiment was completed. The focal length used for the experiment was 210mm. As the size of the aperture increases, the circle of blur increases, so the size of the aperture needed to be as small as possible. This was limited by the camera and lens system to a size of f/32. The focus distance was set to 7 meters which corresponded to the center marker on the structure and the blur on both at the ends of the deck were between 19 and 21 pixels. This allowed for accurate identification of the markers designed for the experiment and all of the markers were visible in the field of view of the camera.

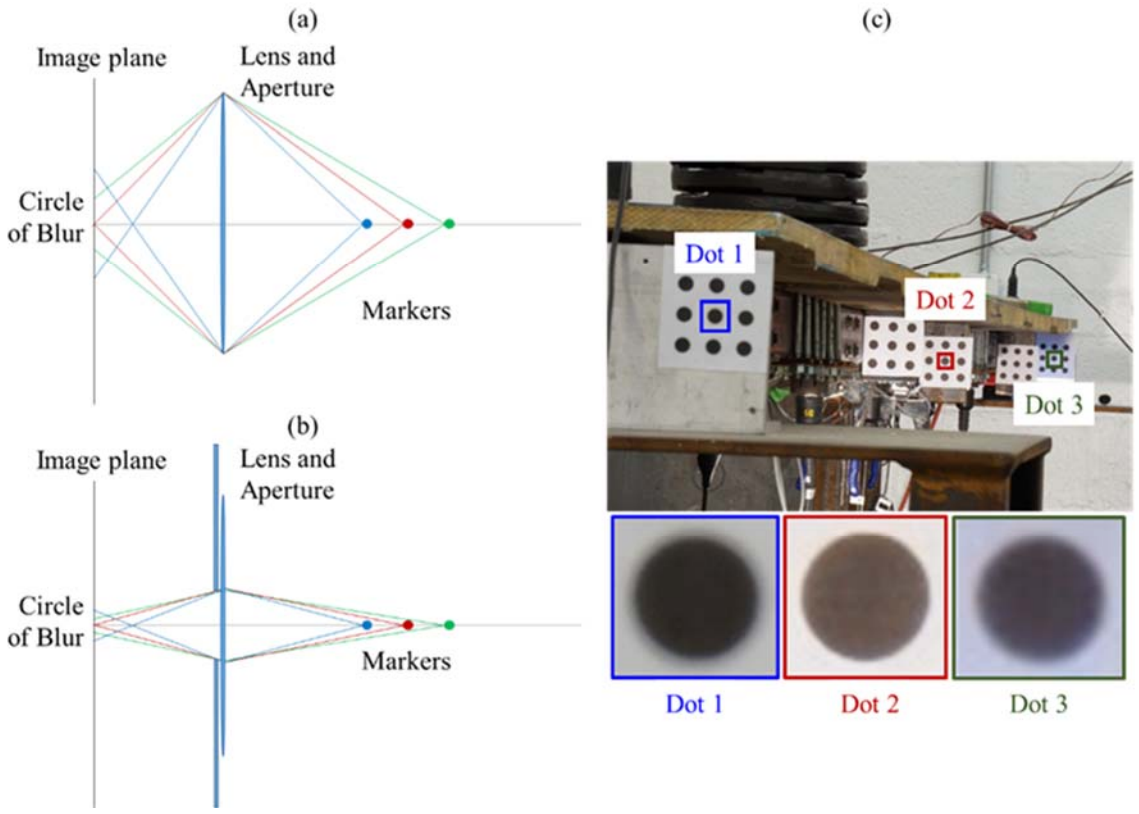


Figure 106. (a) Circle of blur with large aperture. (b) Circle of blur with narrow aperture. (c) Image obtained during the experiment with close up of the end markers and the center marker to highlight the amount of blur.

Figure 106 (a) and (b) show how the amount of blur changes with object distance caused by wide and narrow apertures. The image plane is the location of the camera sensor and from the figure, the circle of blur decreases significantly with a smaller aperture at all of the image planes. Equations (6.7) and (6.8) were used to minimize the amount of image blur at the ends of the deck. The results of the analysis were accurate enough to obtain the results displayed in Figure 106 (c). It is clear from the images that the middle dot is more in focus than the other two. It is also clear that the first and last dot have a similar amount of blur as predicted from the analysis. This is important because they are on opposite sides of the marker in focus so this is the minimum amount of blur possible for all of the markers. If the focus is shifted in either direction, it will cause one of the markers to have a less sharp appearance in the image.

Higher values of ISO correspond to a brighter image and add noise to the image so the ISO setting of the camera was set to be 100 to minimize the amount of noise in the image. The only parameter left to change to increase the brightness of the image was the shutter speed which was set to 4 seconds to obtain a bright enough image. Figure 107 (a) shows the description of the steel grid and where the load was placed. An image taken during the experiment is shown in Figure 107 (b) with all of the markers applied and numbered. In order for the method to be successful in obtaining measurements, all markers had to be within the field of view of the camera for all of the images [294]. Markers 1 and 5 were fixed during the experiment and were used as references to determine how much the other markers moved. They were placed on the supports of the deck to ensure that they did not move throughout the experiment. Markers two, three, and four were fixed to the grid so they would move with the grid while the deck was loaded. Each of these markers

had a string potentiometer displacement gauge placed at the center of the marker for a direction comparison to what was assumed to be ground truth. The displacement gauges were calibrated immediately before testing to ensure they were working and an accurate calibration constant was used for processing. The grid was loaded with 0lbs, 50lbs, 100lbs, 150lbs, and 200lbs and 4 images of each stage were taken at different locations to simulate the movement of a UAS. In order for this method to be implemented on a UAS, a gimbal would need to be used and the shutter speed would need to increase significantly to 1/500s or faster depending upon the setup in order to limit the amount of blur caused by the vibrations of the aircraft.

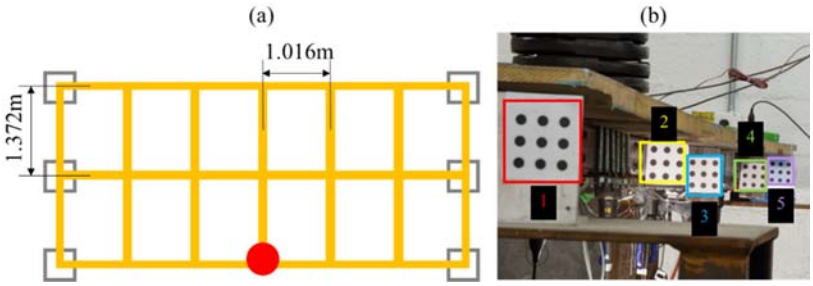


Figure 107. (a) Top view of the steel grid with the load placed at the location of the red circle. (b) Image obtained during the experiment with the markers numbered.

In processing the data, each marker was triangulated using the different locations of the camera. At least two views were needed in order to accomplish this task, as described in section 2.4.5. Assuming the markers do not move along the length of the deck and the motion in the horizontal direction was negligible, equation (2.32) can be altered to obtain better results based on known quantities. The measurements were determined using the fixed markers as a reference and the known locations of the markers along the length of the deck were added into the equation. This added some constraints to make the solution

more accurate. The location of each of the markers is shown in Figure 108 (a). Since this information was used to help constrain the problem, the simplified triangulation was only able to calculate the vertical displacement of the deck. This was not an issue for this experiment because the deck only moved in the vertical direction and the ground truth measurements could only measure the vertical direction as well.

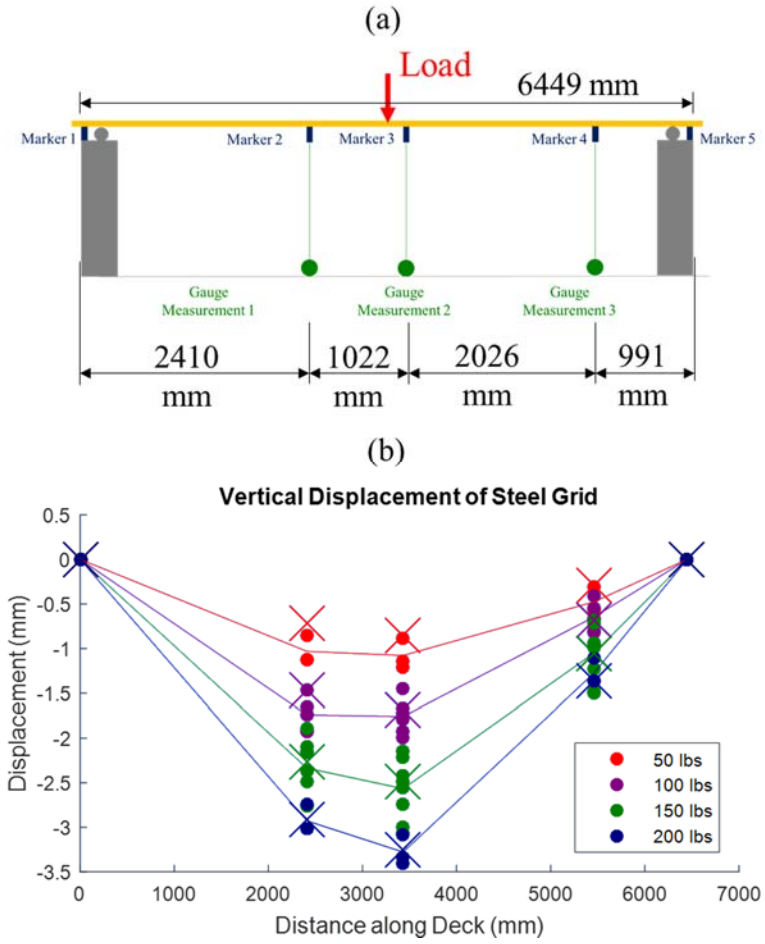


Figure 108. (a) Layout of the markers along the side of the steel grid. (b) Results obtained from the experiment.

The results of the experiment are shown in Figure 108 (b), where the each load stage is color coordinated with the legend and the dots are the image based measurements

and the x's are the displacement gauge measurements. The average of the image measurements taken for each loading stage is represented by a solid line. One advantage of having the fixed markers is that it allows for the assessment of the quality of the measurements. After determining the image measurements, the marker locations were reprojected onto the image and compared to the location of the centroids of each marker using the projection matrices. If the distances from each corresponding centroid were too high, the measurement was removed because of poor triangulation [143]. The numerical values of displacement for the image measurements and the displacement gauges are shown in Table 10.

Table 10. Results of the displacement experiment.

Load (lb)	Mean Displacement Measurements (mm)									
	Marker 1		Marker 2		Marker 3		Marker 4		Marker 5	
	Camera	Gauge	Camera	Gauge	Camera	Gauge	Camera	Gauge	Camera	Gauge
50	0.00	-	1.03	0.72	1.08	0.85	0.48	0.29	0.00	-
100	0.00	-	1.74	1.47	1.76	1.71	0.65	0.69	0.00	-
150	0.00	-	2.34	2.26	2.57	2.48	1.05	1.05	0.00	-
200	0.00	-	2.93	2.92	3.28	3.25	1.28	1.36	0.00	-

The error in millimeters of the results shown in Figure 108 are shown in Table 11 and the percent error is shown in Table 12.

Table 11. Average displacement error results of the steel grid experiment.

Load (lb)	Error (mm)				
50	-	0.31	0.23	0.19	-
100	-	0.28	0.06	-0.04	-
150	-	0.07	0.09	0.00	-
200	-	0.01	0.03	-0.08	-

Table 12. Average percent error results of the steel grid experiment

Load (lb)	Error (%)				
50	-	42.87%	26.69%	65.58%	-
100	-	18.78%	3.73%	-5.97%	-
150	-	3.27%	3.51%	-0.21%	-
200	-	0.43%	0.86%	-5.86%	-

From the tables, it is clear that the average error in millimeters for each stage is less than half a millimeter and in most cases, less than 0.1 millimeters. The percent error appears to decrease as the load increases because of the increasing displacement. A study was done to determine the error associated with the image measurements. The assumption that any markers that had less than one millimeter of reprojection error was a good measurement was enforced and the error associated with the image measurements was determined to be 0.3mm for this experiment. This was determined using two standard deviations of each measurement assuming the error was normal. The major problem associated with this technique is the scale of the experiment heavily affects the outcome and as the scale increases, it becomes impossible to obtain all of the markers in the field of view and achieve enough focus to complete the measurement. Furthermore, as the scale increases, the error associated with each step also increases. Since the error among each marker is minimized to complete the reconstruction, the error is distributed across all of the markers. The minimization takes into account the two view geometry correction which can help, but the underlying problem is still present [143]. One way to mitigate the challenge of scale is to fix the camera and focus on each individual marker, but this is difficult to achieve in the field [294].

6.4 Full Field Beam Displacement Measurement

6.4.1 Multiscale Multispectral Speckle Pattern for 2D DIC

The first multispectral imaging experiments involved creating a pattern that was visible in the infrared spectrum, but not observable in the visible light spectrum. This was accomplished applying multiple types of paint with different emissivity values to a specimen and determining the visibility in the color and IR spectra. The first attempt at accomplishing this goal is shown in Figure 109. Black and white paint as well as clear lacquer were used to determine what was visible in infrared imagery. The clear paint was much more visible in infrared due to the very different emissivity of the aluminum and the lacquer.

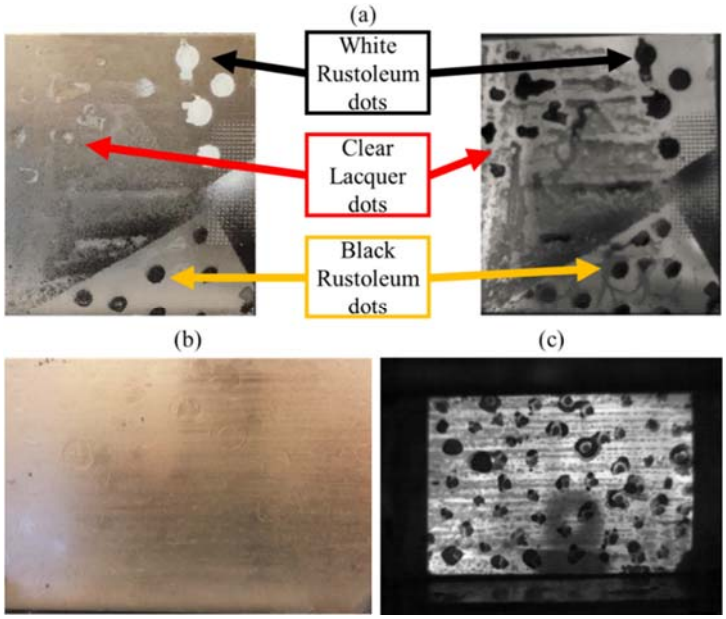


Figure 109. (a) Appearance of different paints in color (left) and infrared imagery (right). (b) Application of the invisible speckle pattern in a color image. (c) Application of the invisible speckle pattern in an infrared image.

After the preliminary testing demonstrating the feasibility of the method, the next experiment focused on DIC with both IR and visible light imagery. Black point was used

to create the speckles so they would be visible in both visible and infrared imagery. Images were taken of a beam in the lab which was loaded and DIC was used to determine the displacement. These displacements were then compared to a displacement gauge. An example of each of the color and infrared images taken during the experiment are shown in Figure 110 (a) and (b) respectively and the results of the experiment are shown in Figure 110 (c). For both loads, the displacement gauges and the DIC measurements were almost identical with less than 1% error with respect to the displacement gauges.

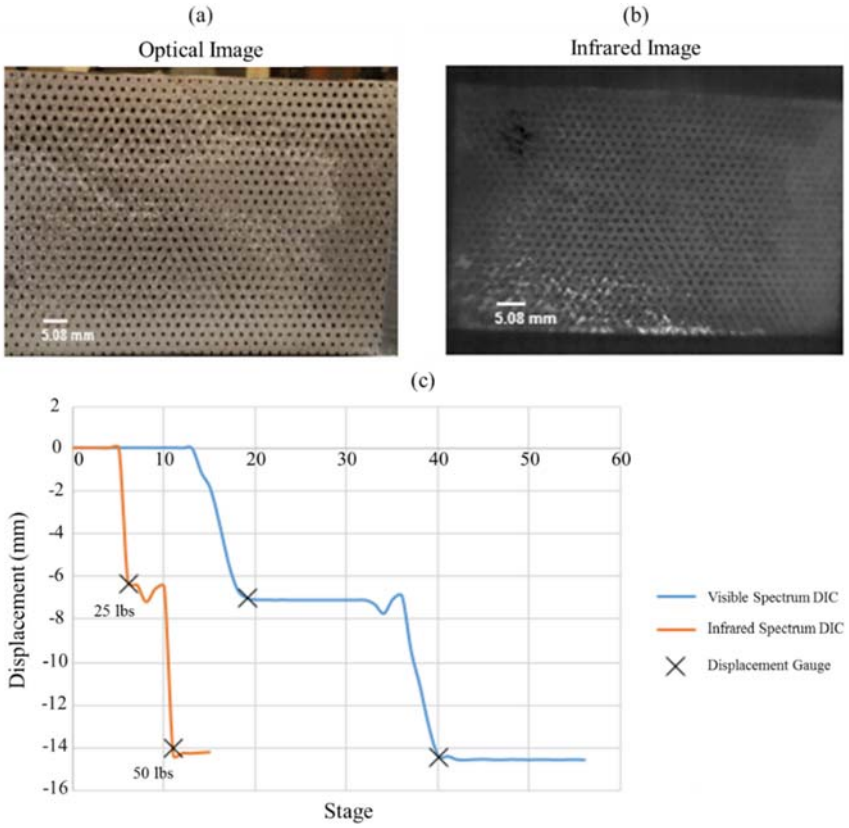


Figure 110. (a) Color image from the experiment. (b) Infrared image from the experiment. (c) Displacement of the structure measured by visible light DIC, infrared DIC, and displacement gauges.

After validating the results of IR DIC with traditional DIC, more careful consideration was taken to determining the wavelength to achieve better results. The

resolving power, described in section 2.5.2, was considered to determine the wavelength that would be used in the next experiments. In addition to the resolving power, the optimal size of the speckles for a specified distance, focal length, sensor size, and resolution was determined for multiple distances.

A 24 megapixel Sony a6000 camera with a CMOS 23.5mm by 15.6mm sensor, Bayer filter, and 16mm focal length was used to capture color images during this experiment. The angle per pixel was determined using equation (9) and (10) where θ_p is the angle per pixel in radians, s_x and s_y is the horizontal and vertical size of the sensor in millimeters, f is the focal length in millimeters, and N_x and N_y is the number of pixels in the horizontal and vertical direction [295].

$$\theta_p = \frac{\arctan\left(\frac{s_y}{2f}\right)}{\frac{N_y}{2}} \tag{9}$$

$$\theta_p = \frac{\arctan\left(\frac{s_x}{2f}\right)}{\frac{N_x}{2}} \tag{10}$$

The angle per pixel is roughly the same in both the horizontal and vertical directions. The average of these values were then used to determine the size of the speckles (S) based on the object distance (O) and the number of pixels per speckle (n) using equation (11). This quantity was critical to the design of the speckle pattern and several images of a crack card were taken at multiple distances and analyzed to determine the smallest size resolvable with a specified number of pixels to validate the equation. The minimum size of the speckle for registration is 3x3 pixels; therefore, n was set to be 3. The images taken at multiple distances and results are shown in Figure 111.

$$S = \tan\left(n \cdot \theta_p\right) \cdot O \tag{11}$$

A diagram describing the variables in equation (11) is shown in Figure 111 (a). The images in Figure 111 (b) were taken at 6, 12, 18, and 24 inches from the crack card. The pixel size was then compared to the results of equation (11). The smallest resolvable line width is shown in Figure 111 (c) and (d) with the red box corresponding to a 3x3 pixel neighborhood, the smallest acceptable size of the speckle.

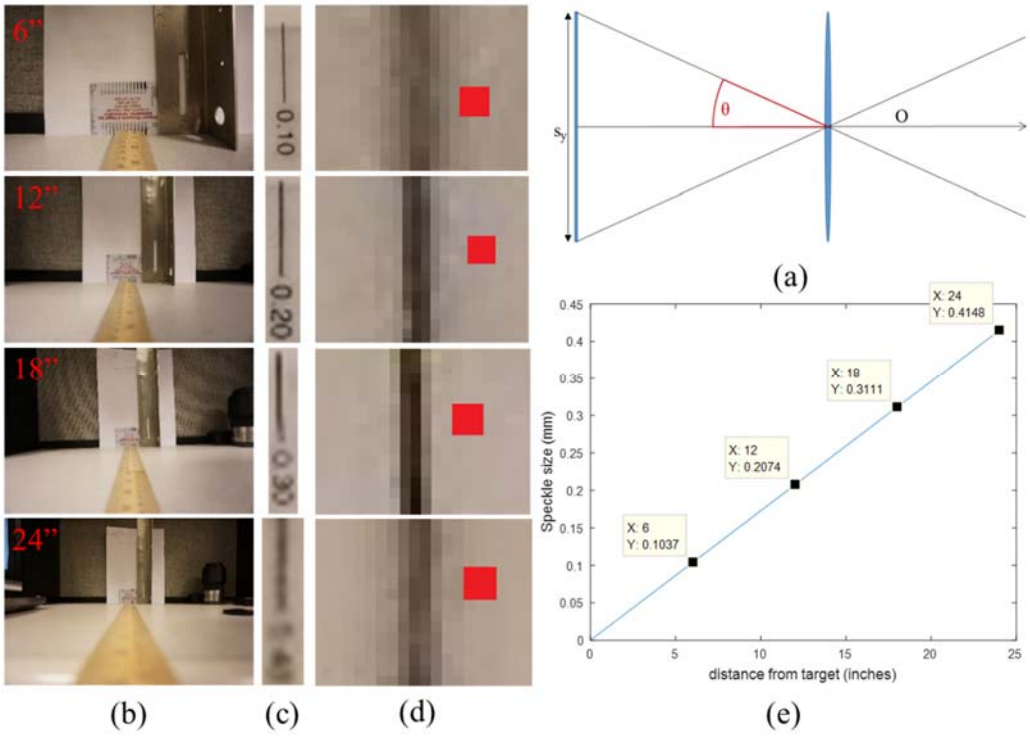


Figure 111. (a) Diagram describing equations (9), (10), and (11). (b) Images taken of a crack card with predetermined widths. (c) Expanded version of the smallest size the camera can resolve. (d) Further expansion to show a 3x3 pixel neighborhood in red. (e) Comparison of the values obtained with the equation at each of the distances shown in (b).

Figure 111 (e) validates that the theoretical equations and the experimental results correlate well for the camera lens system. The numerical results of the experiment are

shown in Table 13 where the image speckle size was determined manually from Figure 111 (d).

Table 13. Results of validation experiment shown in Figure 111.

Distance (inches)	Image Speckle Size (mm)	Theoretical Speckle Size (mm)
6	0.1	0.1037
12	0.2	0.2074
18	0.3	0.3111
24	0.4	0.4148

The equations were then used to determine the optimal speckle size given distance. In addition to the speckle size, the distribution of points was a key parameter to creating a good speckle pattern. There are several metrics that can be used to determine the global and local quality of a speckle pattern. The ones chosen for this experiment were the local sum of squares of subset intensity gradient which was validated against the local subset entropy and the global mean intensity gradient [184]. Simulated images were used to compare these metrics and were then used to create a stereolithography (STL) file used to 3D print a stencil. The validation is shown in Figure 112. As the mean intensity gradient increases, the local metrics also increase resulting in a better speckle pattern. The higher values in both metrics indicate a greater amount of texture which is better for DIC algorithms to track.

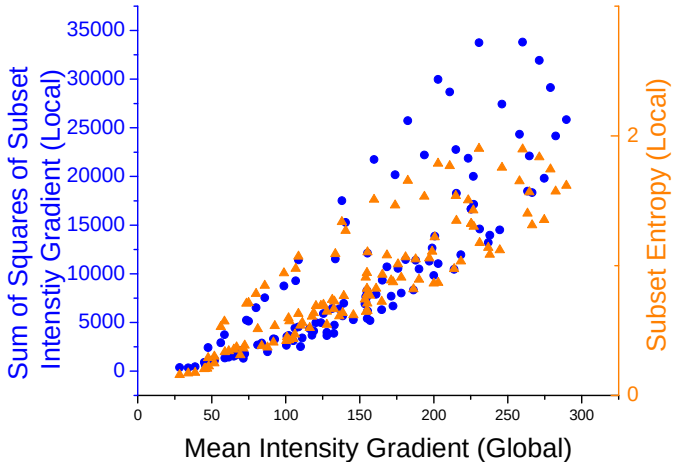


Figure 112. Comparison of global and local speckle pattern metrics

The plots in Figure 113 show the how the particle area and number of particles affect the mean intensity gradient values. Smaller particles and a greater number of particles result in higher MIG values. Therefore, an optimal pattern should contain the smallest speckles possible and as many particles as possible. However, the minimum size of the particle is limited to 3 pixels since fewer pixels result in aliasing which causes failure in particle tracking which results in a loss of facets. Furthermore, the speckles within each facet must be unique within a predefined search radius to ensure proper tracking of each facet.

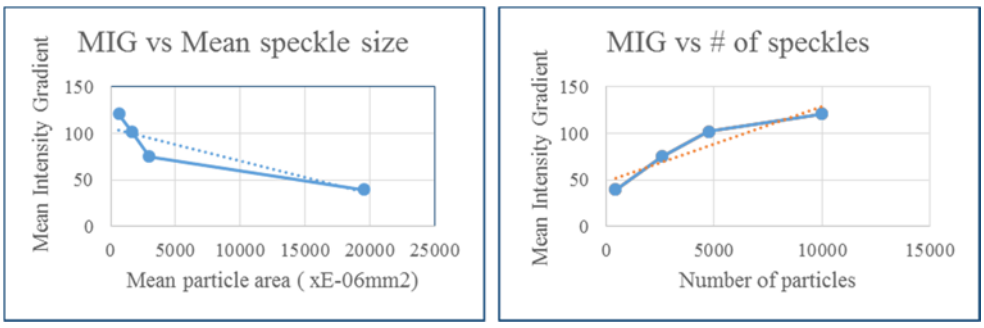


Figure 113. Comparison of speckle size vs. speckle quality metrics

After the optimal speckle properties were determined, a template was manufactured with a 3D printer for each of the speckle sizes. The minimum print resolution of the 3D printer was used as to determine the smallest printable speckle size. Several STL files were created from the images created in MATLAB with different sizes and each pattern was printed with the 3D printer. The top images in Figure 114 show the patterns designed in MATLAB and the corresponding bottom images show the resulting 3D printed template. It was determined that the smallest acceptable speckle size was 1mm diameter circles based on the template quality. In order to ensure that the stencil was able to be manufactured at the proper size, the smaller dot size was set to 2mm. The larger dot size was determined based on the smaller pattern. Since the smallest acceptable speckle size is 3x3 pixels, the larger speckles were designed to have 6mm diameter circles. It was expected that the smaller speckles would be visible at the farther distance, but heavily affected by aliasing.

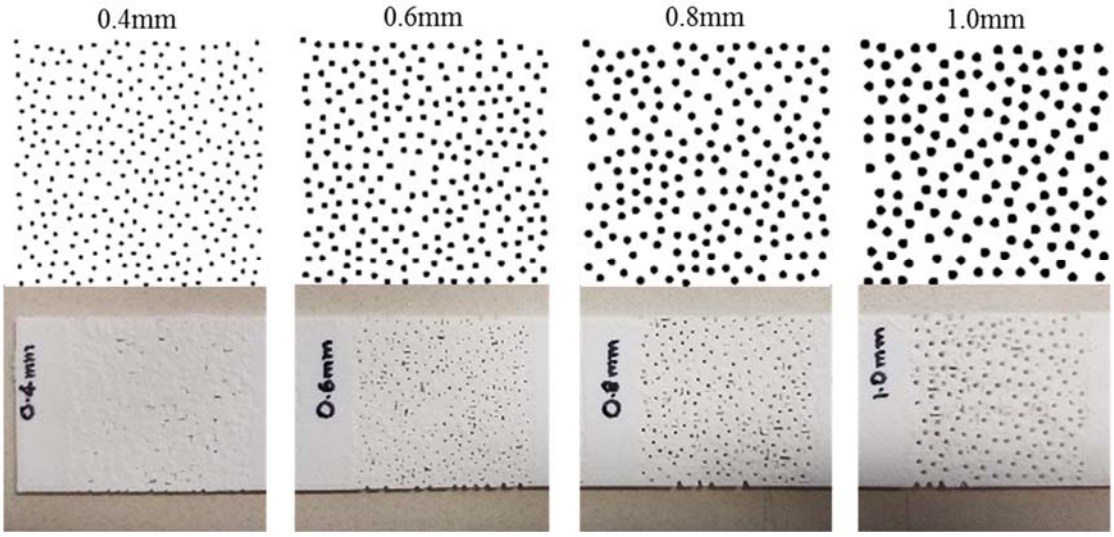


Figure 114. Computer generated speckle templates used to determine the minimum printable dot size for the stencils

The design for the experiment is shown in Figure 115. The two different template sizes were optimized for the two corresponding distances. The horizontal field of view is represented by the angle θ which remains constant for both the near and far distances. Both patterns were spatially optimized to ensure the 3D printer would accurately print the dots with no overlap in addition to ensuring the speckles produced high values for the MIG for the optimal distance.

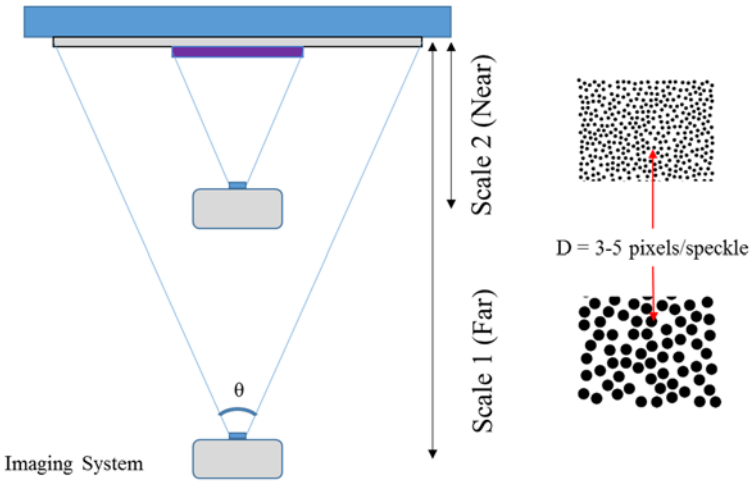


Figure 115. Generation of pseudo-random, optimized speckle pattern for predetermined working distances [227]

The images formed in MATLAB were used to form an STL file by assigning a thickness and the files were then sent to the 3D printer for manufacturing. The final stencils used for the experiment are shown in Figure 116 (a). The larger pattern was applied to the structure using black and white spray paint. The beam was first coated in white paint and after it was dry, the stencil was positioned on the beam and black spray paint was applied. After the black paint had dried, an invisible, non-reflective paint was applied so that when a black light was placed near the beam, the paint would not appear to fluoresce. After that

coat dried, the smaller stencil was placed on the structure and an airbrush was used to apply the clear paint with pigment inside that appeared bright when the black light was put near the structure. The patterns are shown in Figure 116 (b) after they were applied to the beam.

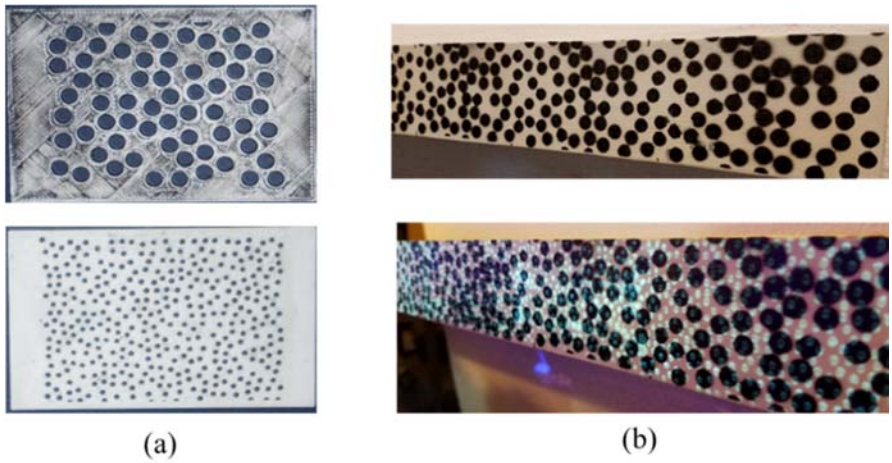


Figure 116. (a) Manufactured pattern stencils, and (b) applied patterns on the beam

The top image in Figure 116 (b) shows the color image with both patterns applied to it without the black light being present. The clear paint is not visible in this condition. The lower image shows the same beam with the smaller speckle pattern visibly fluorescing with a black light near it.

A finite element model was created of the beam prior to the experiment. The model was formed with brick elements and assumed to be pinned at two locations on the structure to be more accurate to the actual boundary condition where the beam was clamped to the support. The model is shown in Figure 117 with the assumed loading shown in (a) and the results shown in (b). The main purpose of the model was to ensure that the beam would not be damaged under the loading and for comparison to the actual results obtained from DIC and the displacement gauges.

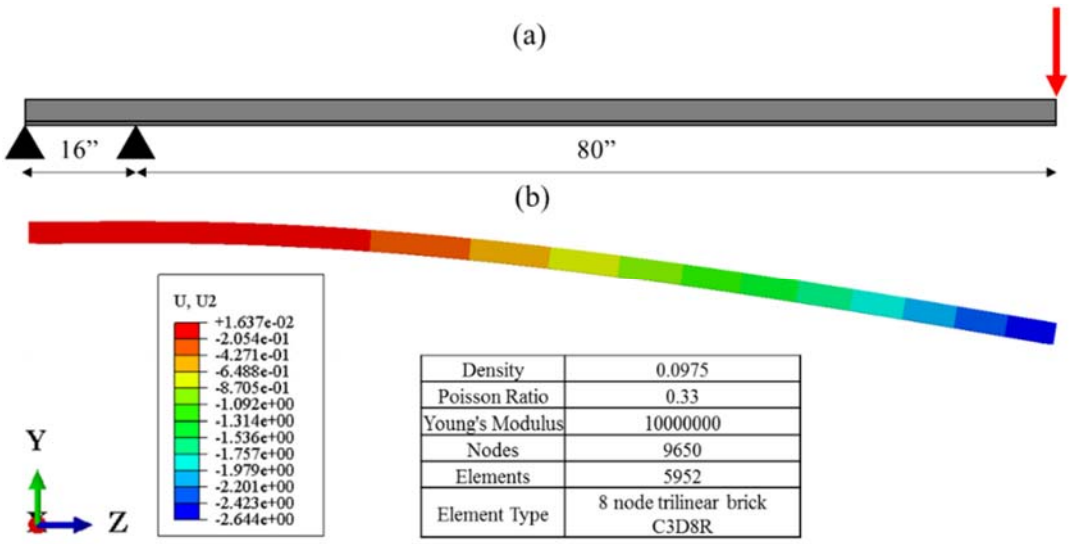


Figure 117. (a) Assumed loading for the model. (b) Results of the finite element model and properties

The model for plasticity of Al-6061-T6 used for the model is tabulated below [296].

Table 14. Stress and strain for the material law used in the model in Figure 117

Stress (psi)	Strain (%)
41,480.868	0.384
42,061.020	0.438
43,221.324	0.493
43,801.476	0.651
44,236.590	0.992
44,816.742	2
45,686.970	3.51
46,267.122	5.02
46,702.236	6.51
46,847.274	7.44

An aluminum C-beam with the cross section shown at the top of Figure 118 (a) was chosen as the specimen for the experiment. This beam was chosen because it was not

expected to have any out of plane motion due to the symmetry. The beam was fixed at one end and loaded on the other. The speckle patterns were applied near the loading side. Three displacement gauges were used as the ground truth and placed at locations along the beam for direct comparison to points within the full field DIC measurements. The setup and dimensions of the beam are shown in Figure 118 (a) including the locations of the displacement gauges. Lights were applied to the beam to ensure the lighting was as uniform as possible and did not change throughout the experiment. The only camera setting that was changed during the experiment when switching from the standard color images to the images capture with a present black light was the exposure time. This ensured that the depth of field, noise properties, and magnification from the lens did not change both sets of images. This made the measurements directly comparable because the exposure time should have the least effect on the quality of the images, aside from changing the brightness. Table 15 shows the camera parameters for the color and UV patterns during the experiment.

Table 15. Camera Settings during the experiment

Camera Parameters	Color	UV
Aperture	f/22	f/22
Exposure Time	0.5s	4s
ISO	100	100
Focal Length	16mm	16mm

The images shown in Figure 118 (b) and (c) were taken during the experiment at the closer distance with and without the black light applied respectively. The speckles

appear to be blue when the black light was applied and were not visible when using the standard lighting conditions.

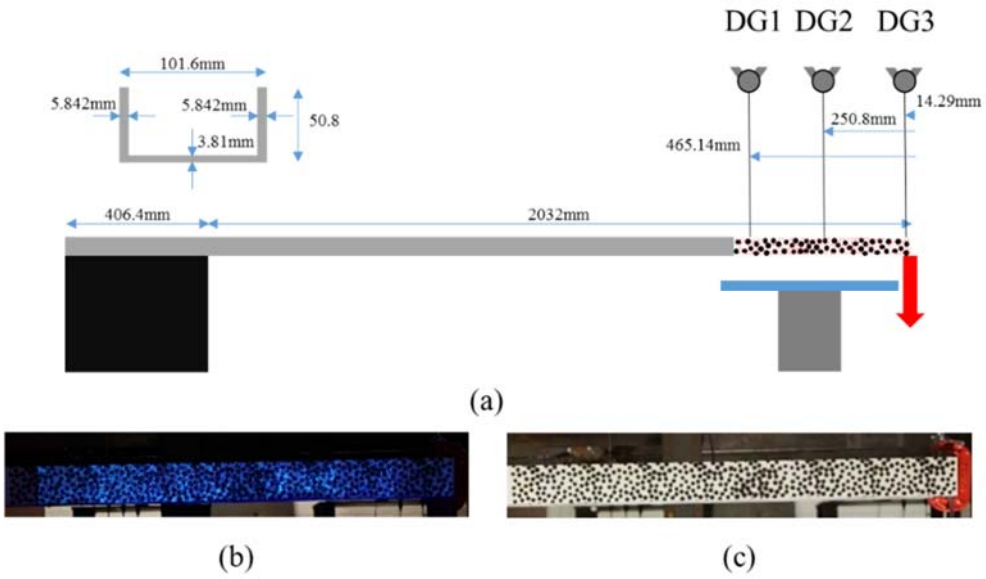


Figure 118. (a) Experimental Setup. Images of the (b) UV pattern and (c) color pattern obtained during the experiment.

Since 2D DIC was used to process the data obtained from this experiment, the distance per pixel was required to convert the pixel measurements into distance measurements. Calibration images were taken of a calibration panel prior to the experiment with known distances between each marker. The assumption with the 2D parameter is that the camera would remain stationary throughout the experiment. Figure 119 shows one of the calibration images taken prior to the experiment. The calibration images were taken at each predetermined distance in the exact same location as the images were taken during the experiment. To ensure the camera didn't move during the experiment, the camera was tethered and triggered by a computer so the camera did not have to be touched after it was placed on the tripod and the calibration images were taken.

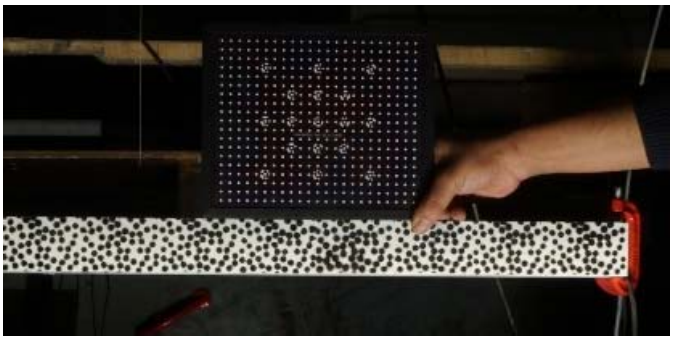


Figure 119. Calibration panel used to determine distance per pixel.

The calibration panel was aligned with the front face of the beam to ensure the calculated 2D parameter was correct for the DIC algorithm. If this was not the case, large errors would result from an incorrect 2D parameter. The points of on the calibration panel were well known and the images were processed to isolate the dots in the calibration panel as shown in Figure 120. This process was done at both distances to calculate the 2D parameter.

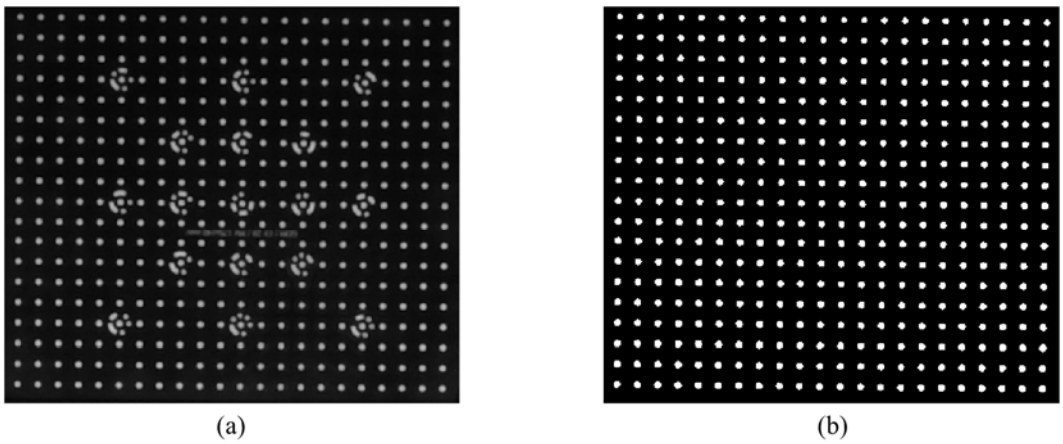


Figure 120. (a) Original image of the calibration panel. (b) Processed image to isolate the dots.

The segmentation of the dots was accomplished with thresholding using the Otsu method, followed by Hough transform used to identify lines across the image, and finally selecting only blobs that existed at the intersections of the lines. This ensured that only the known points were identified and rejected the other areas of the image corresponding to coded markers in the calibration panel that did not have a well-known location.

Each of the points on the calibration panel with well-known locations with respect to each other were used to calculate the 2D parameter required for 2D DIC. The large number of points in the panel provided a higher degree of confidence in the distance per pixel. Using this technique, the 2D parameter at the closer distance was determined to be 0.46mm/pixel and 1.38mm/pixel at the farther distance. This technique was effective in determining the 2D parameter as well as determining if the camera image plane was not parallel to the structure.

Five images of each stage were taken and averaged to reduce the noise associated with the camera system prior to analyzing the images with DIC. ARAMIS DIC software was used to compute the displacements. ARAMIS only accepts grayscale images to compute the displacements [288]. However, due to the design of the Bayer filter, the green band of the image has the highest amount of spatial content [154]. Therefore, rather than simply averaging the color bands to obtain a grayscale image, the green band was isolated and processed so the highest spatial content was used and information was not averaged across the bands with less spatial content. Ideally, it would be beneficial to use a monochrome camera with corresponding filters to ensure only the desired wavelengths of light were visible. This would ensure the image takes full advantage of all of the pixels rather than interpolating between neighboring pixels of the same color. It could also

eliminate the need for the active lighting with the black light. The facet size used to post-process the images obtained at the closer distance was 21 pixels with a spacing of 10 pixels. This ensured a significant amount of overlap of the facets so that light intensity changes would be picked up by neighboring facets to obtain good measurements. The facet size used to post-process the images taken that the farther distance was 11 pixels and a 5 pixel spacing. The measurements obtained at the 2m working distance are in shown in Table 16 and Table 17 and the results are displayed in Figure 121 (a) and (b) for color and UV imagery respectively.

Table 16. Displacement measurements from color images at the 2m working distance.

Close (2m)	Displacement Gauges (mm)			Color Measurement (mm)		
load (kg)	DG1	DG2	DG3	DG1	DG2	DG3
Baseline	0	0	0	0	0	0
0	0.17	0.10	0.13	0.20	0.14	-0.04
11.34	-15.30	-18.71	-22.50	-15.77	-19.00	-22.81
11.34	-14.83	-18.12	-21.83	-15.30	-18.46	-22.23
22.68	-30.66	-37.58	-44.80	-30.75	-37.07	-44.54
22.68	-30.10	-36.86	-44.05	-30.45	-36.72	-44.11
45.36	-46.57	-56.87	-67.53	-45.44	-54.81	-65.92
45.36	-46.35	-56.64	-67.28	-45.57	-54.95	-66.09

Table 17. Displacement measurements from UV images at the 2m working distance.

Close (2m)	Displacement Gauges (mm)			UV Measurement (mm)		
load (kg)	DG1	DG2	DG3	DG1	DG2	DG3
Baseline	0	0	0	0	0	0
0	-0.08	-0.10	-0.11	0.01	0.02	0.05
11.34	-14.93	-18.24	-21.95	-15.21	-18.33	-22.00
11.34	-15.38	-18.78	-22.59	-15.51	-18.71	-22.37
22.68	-30.32	-37.15	-44.34	-30.17	-36.44	-44.07
22.68	-30.73	-37.65	-44.88	-30.58	-36.84	-44.45
45.36	-46.43	-56.73	-67.38	-45.33	-54.63	-65.94
45.36	-46.42	-56.65	-67.35	-45.35	-54.68	-66.04

The measurements obtained at the 6m working distance are in Table 18 and Table 19 and displayed in Figure 122 (a) and (b) for color and UV imagery respectively.

Table 18. Displacement measurements from color images at the 6m working distance.

Far (6m)	Displacement Gauges (mm)			Color Measurement (mm)		
load (kg)	DG1	DG2	DG3	DG1	DG2	DG3
Baseline	0	0	0	0	0	0
0	-0.14	-0.22	-0.11	-0.45	-0.44	-0.50
11.34	-14.91	-18.15	-21.88	-15.40	-18.69	-22.52
11.34	-15.29	-18.65	-22.41	-16.38	-19.73	-23.67
22.68	-30.21	-36.96	-44.14	-30.97	-37.36	-44.96
22.68	-30.97	-37.97	-45.12	-31.58	-38.01	-45.67
45.36	-46.52	-56.84	-67.43	-46.49	-56.07	-67.38
45.36	-46.61	-56.85	-67.54	-46.60	-56.13	-67.42

Table 19. Displacement measurements from UV images at the 6m working distance.

Far (6m)	Displacement Gauges (mm)			UV Measurement (mm)		
load (kg)	DG1	DG2	DG3	DG1	DG2	DG3
Baseline	0	0	0	0	0	0
0	-0.06	-0.13	-0.19	-0.12	-0.09	-0.01
11.34	-15.54	-19.01	-22.80	-13.85	-13.91	-17.79
11.34	-14.96	-18.24	-22.05	-14.90	-18.05	-22.94
22.68	-30.89	-37.84	-45.04	-30.99	-38.49	-44.86
22.68	-30.36	-37.15	-44.43	-30.52	-37.93	-44.19
45.36	-46.60	-56.92	-67.58	-46.03	-56.66	-66.81
45.36	-46.64	-56.94	-67.65	-45.84	-56.64	-66.55

The x markers correspond to the image based measurements at the location of the displacement gauges extracted from the calculated displacement field. The lines correspond to the displacement gauge measurements which were assumed to be the ground truth measurements for the experiment.

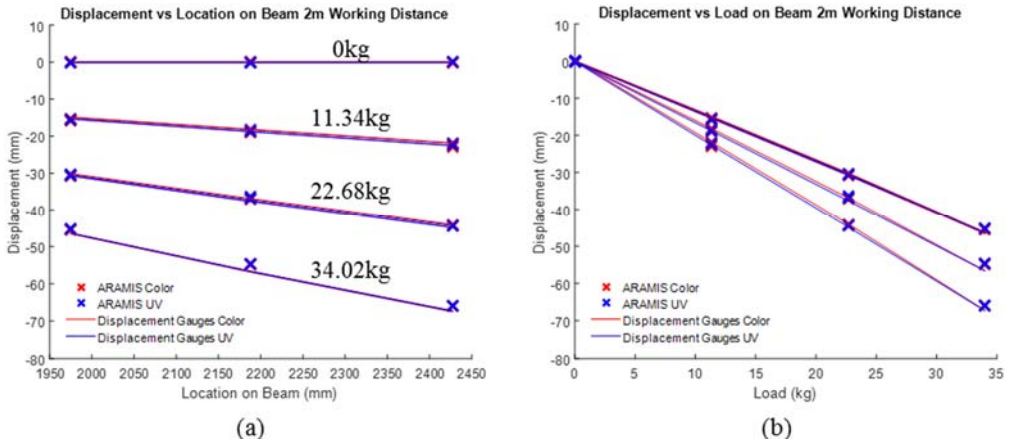


Figure 121. Results of displacement gauges and ARAMIS DIC data at points corresponding to the displacement gauges. (a) Deformation vs. location on the beam at 2m working distance. (b) Deformation vs. load at 2m working distance.

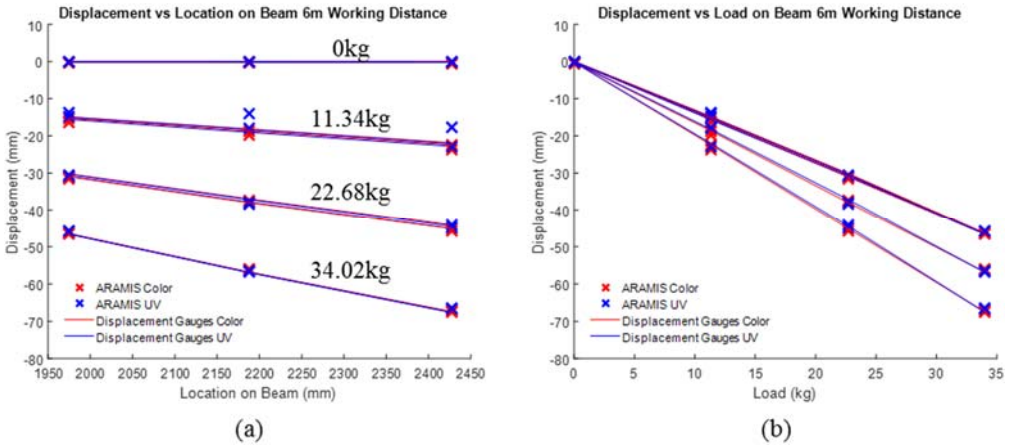


Figure 122. Results of displacement gauges and ARAMIS DIC data at points corresponding to the displacement gauges. (a) Deformation vs. location on the beam at 6m working distance. (b) Deformation vs. load at 6m working distance.

The localization of the displacement gauges within the full field measurements was difficult due to the size of the facets, spacing of the facets, and the aliasing that occurs at the edges of the beam. Therefore, the end of the beam was used as a reference to determine the approximate location of the displacement gauges and the facet closest to this location was compared to the displacement gauges. The results in Figure 121 (a) and (b) correspond

to the 2 meter working distance while Figure 122 (a) and (b) correspond to the 6 meter working distance. Figure 121 (a) and Figure 122 (a) show the displacement of the beam vs. the position along on the beam and Figure 121 (b) and Figure 122 (b) show the displacement vs. loading.

It is clear from the plots that the measurements obtained from ARAMIS performed well when compared with the displacement gauges. The errors in millimeters are shown in Table 20 and Table 21 and the percent errors are shown in Table 22 and Table 23.

Table 20. Error of the ARAMIS data with respect to the displacement gauges in millimeters for the 2m working distance

load (kg)	Error (mm) Color Close			Error (mm) UV Close		
	DG1	DG2	DG3	DG1	DG2	DG3
Baseline	0	0	0	0	0	0
0	-0.03	-0.04	0.17	-0.09	-0.12	-0.16
11.34	0.47	0.29	0.31	0.27	0.09	0.06
11.34	0.48	0.34	0.40	0.13	-0.06	-0.23
22.68	0.09	-0.51	-0.26	-0.15	-0.71	-0.27
22.68	0.35	-0.14	0.06	-0.15	-0.81	-0.43
34.02	-1.13	-2.07	-1.61	-1.10	-2.11	-1.44
34.02	-0.79	-1.69	-1.19	-1.07	-1.97	-1.31

Table 21. Error of the ARAMIS data with respect to the displacement gauges in millimeters for the 6m working distance

load (kg)	Error (mm) Color Far			Error (mm) UV Far		
	DG1	DG2	DG3	DG1	DG2	DG3
Baseline	0	0	0	0	0	0
0	0.31	0.22	0.39	0.06	-0.05	-0.17
11.34	0.49	0.54	0.65	-1.69	-5.10	-5.01
11.34	1.09	1.08	1.25	-0.06	-0.20	0.89
22.68	0.76	0.40	0.82	0.11	0.65	-0.18
22.68	0.61	0.05	0.55	0.15	0.78	-0.24
34.02	-0.03	-0.77	-0.06	-0.56	-0.26	-0.77
34.02	-0.01	-0.72	-0.12	-0.81	-0.30	-1.11

Table 22. Percent error of the ARAMIS data with respect to the displacement gauges for the 2m working distance

load (kg)	Error (%) Color Close			Error (%) UV Close		
	DG1	DG2	DG3	DG1	DG2	DG3
Baseline	-	-	-	-	-	-
0	-15.0%	-43.6%	130.8%	111.8%	117.5%	142.33%
11.34	-3.1%	-1.6%	-1.4%	-1.8%	-0.5%	-0.3%
11.34	-3.21%	-1.86%	-1.84%	-0.87%	0.34%	1.00%
22.68	-0.3%	1.4%	0.6%	0.5%	1.9%	0.6%
22.68	-1.2%	0.4%	-0.1%	0.5%	2.2%	1.0%
34.02	2.4%	3.6%	2.4%	2.4%	3.7%	2.1%
34.02	1.7%	3.0%	1.8%	2.3%	3.5%	2.0 %

Table 23. Percent error of the ARAMIS data with respect to the displacement gauges for the 6m working distance

load (kg)	Error (%) Color Far			Error (%) UV Far		
	DG1	DG2	DG3	DG1	DG2	DG3
Baseline	-	-	-	-	-	-
0	-226.2%	-97.9%	-355.7%	-92.9%	35.3%	93.4%
11.34	-3.28%	-3.0%	-3.0%	11.0%	26.8%	22.0%
11.34	-7.14%	-5.79%	-5.58%	0.38%	1.08%	-4.04%
22.68	-2.5%	-1.1%	-1.9%	-0.3%	-1.7%	0.4%
22.68	-2.0 %	-0.1%	-1.2%	-0.5%	-2.1%	0.5%
34.02	0.1%	1.4%	0.1%	1.2%	0.5%	1.1%
34.02	0.0%	1.3%	0.2%	1.7%	0.5%	1.6%

The errors are, in almost all cases, within 5% of the displacement gauges with the exception of the 0lb load that was not the baseline because the values should have been zero. One of the major sources of error that contribute to the error is the 2D parameter set in ARAMIS to calculate the deformation in millimeters from the deformation in pixels because even with the calibration images, small errors in this value result in larger errors in displacement, particularly for the later stages. In addition, the exact locations of the displacement gauges were difficult to accurately determine so the closest facet centroid

was used, which could also contribute to some of the errors. Furthermore, the camera was assumed to be fixed during the experiment. If it moved, this would be would cause a significant amount of error, particularly for the farther distance. Though the camera was on a tripod, the camera shutter may have caused slight movements.

Due to the size requirements of the speckle patterns, it was not expected that the errors would be as small as in other DIC applications because of the large size of the pixels and the noise floor associated with a CMOS color camera. Even for the close distance, the size of the pixel was just under half a millimeter. However, Figure 123 proves that the speckle patterns that were optimized for each distance had a lower spread for the noise than the other pattern that was not optimized for that distance.

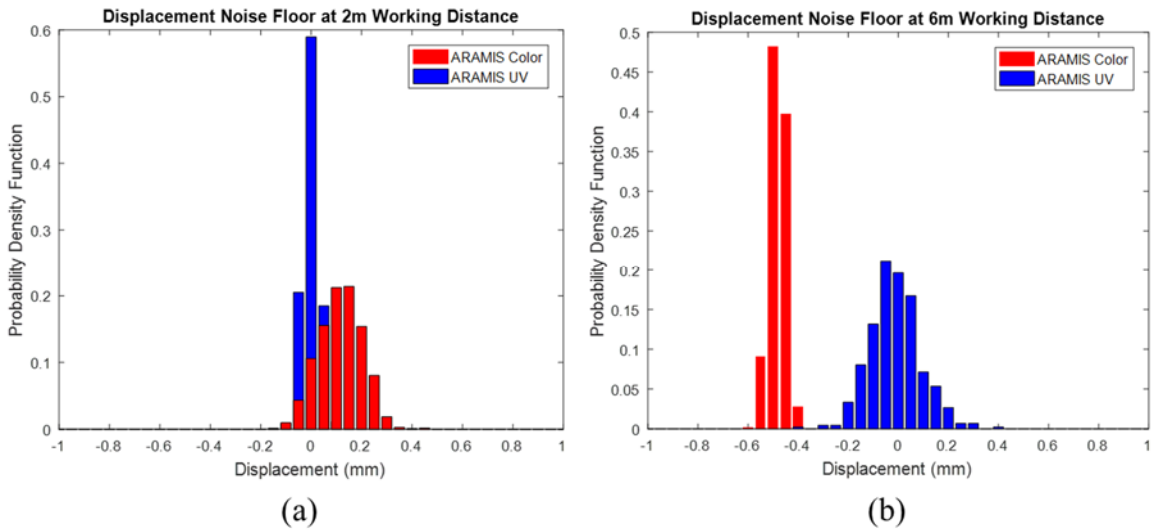


Figure 123. (a) Noise floor at close distance. (b) Noise floor at far distance.

Figure 123 (a) shows the noise distribution of the displacement with zero load at the 2 meter distance. It is clear that the UV pattern shown in blue has a smaller spread than the color pattern shown in red. Similarly, Figure 123 (b) shows the distribution of the

displacement noise with zero load at the 6 meter distance. It is clear that the UV pattern shown in blue has a larger spread than the color pattern shown in red. This demonstrates that both patterns that were designed for a specified distance performed better than the pattern that was not designed for that distance. The mismatch in the center of the distributions in Figure 123 is due to the beam not returning to the original position due to imperfect clamping on the fixed end of the beam. The beam was not plastically deformed during the experiment based on the stresses observed in the finite element model shown in Figure 117.

6.4.2 Measurement from UAS Imagery

The next experiment was designed to validate that 2D DIC can be used in conjunction with a UAS to obtain full field displacement measurements. The lab specimen used during the test was a cantilever aluminum L-beam. Figure 124 (a) shows the diagram of the setup and (b) shows an image of the setup with the fixed end secured by C-clamps. The fixed end is shown in purple, the speckle pattern is shown in red, and the load is shown in yellow. The string potentiometers are displayed in green with lines showing the locations of the measurement control points used during the experiment. The string potentiometers were assumed to be the ground truth in which the full field image measurements were compared. Since the string potentiometers only give point measurements, these points were compared to the closest facet centroids at the same locations in the full field measurements. The displacement gauges were fixed to the front of the beam so they were easier to identify their locations in the images. The speckles were very large in this experiment because the position was not able to be well controlled with the UAS. This ensured that no matter what

the distance, the pattern could be tracked, even if it was not optimized using the method described in section 6.4.1

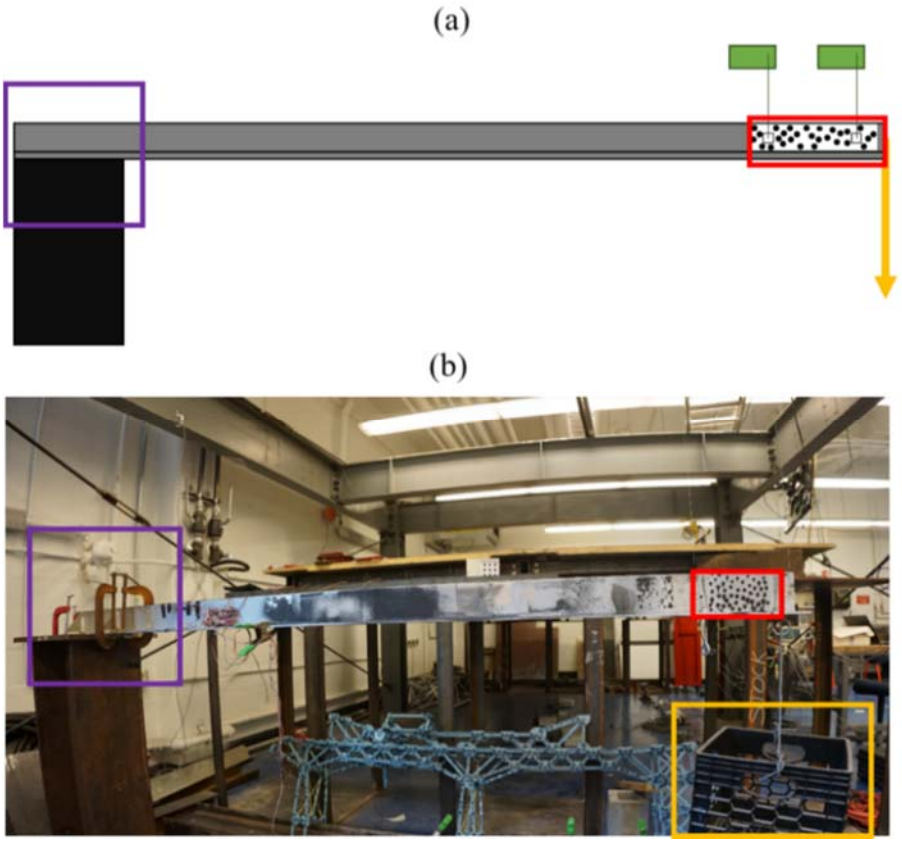
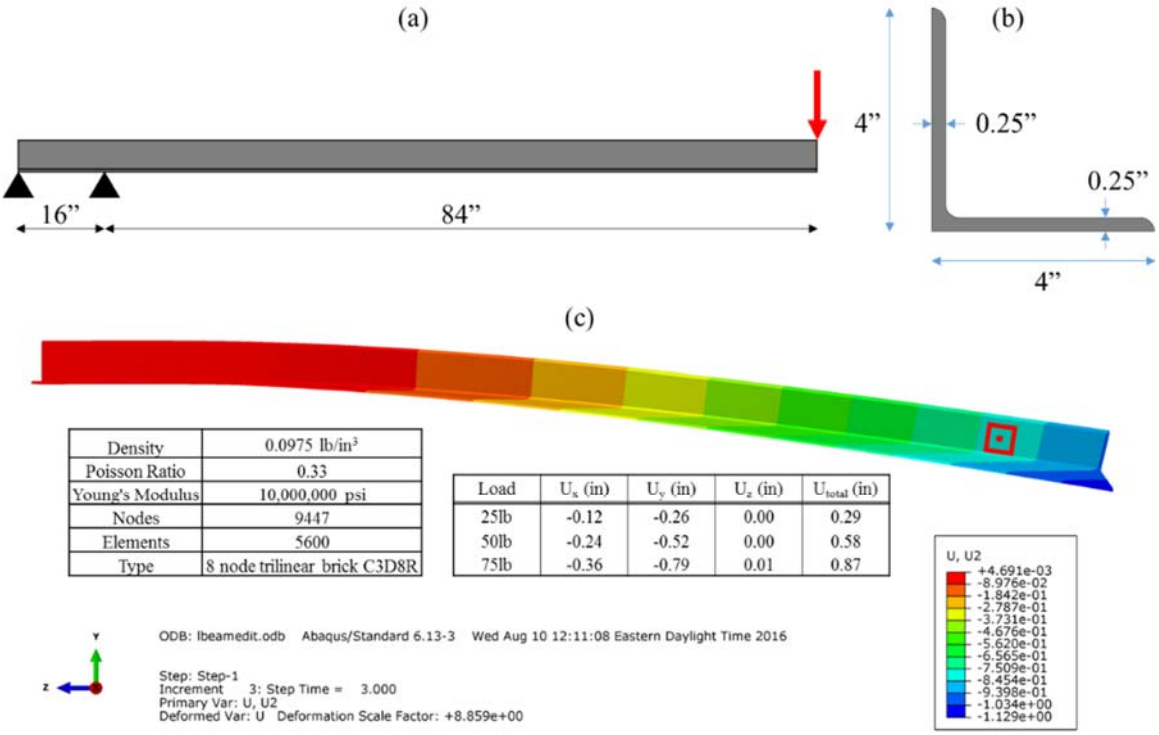


Figure 124. (a) Diagram of experimental setup. (b) Picture of the experimental setup.

A finite element simulation was completed in ABAQUS for the comparison of the results and validation of movement in the out of plane direction. Since the beam was asymmetric, the beam was expected to have out of plane motion that could negatively impact the DIC measurements. The results of the ABAQUS model demonstrated that the out of plane motion could be neglected given the camera and lens combination used for the experiment. The load diagram and the cross section of the beam are shown in Figure 125 (a) and (b), Figure 125 (c) shows the results and properties of the model. The red dot on

the beam is the approximate location in the model of the displacement gauge used as the ground truth for these measurements.



Density	0.0975 lb/in³
Poisson Ratio	0.33
Young's Modulus	10,000,000 psi
Nodes	9447
Elements	5600
Type	8 node trilinear brick C3D8R

Load	U_x (in)	U_y (in)	U_z (in)	U_{total} (in)
25lb	-0.12	-0.26	0.00	0.29
50lb	-0.24	-0.52	0.00	0.58
75lb	-0.36	-0.79	0.01	0.87

Figure 125. (a) Load diagram and boundary conditions assumed for the finite element model. (b) Cross section of the beam. (c) Displacement results of the finite element simulation and finite element properties.

The boundary conditions for the finite element model were chosen based on the locations of the C-clamps in the experiment. In addition to the errors associated with out of plane motion for 2D DIC, the out of plane motion was expected to add error to the displacement gauge measurements, however the effect on these measurements is also negligible because the sensors were placed far away from the beam with strings connecting the sensor to the beam. This caused a very small angle associated with the string making the effect of the out of plane motion negligible for this system as well.

The first measurements were conducted with a high resolution camera placed on a tripod. The camera was tethered to a computer and triggered remotely to ensure the camera did not move throughout the experiment. DIC was used to track the deformation of the beam using Ncorr in MATLAB [297, 298]. The beam was loaded and the Ncorr results are displayed in Figure 126 for 25lb, 50lb, and 75lb loads.

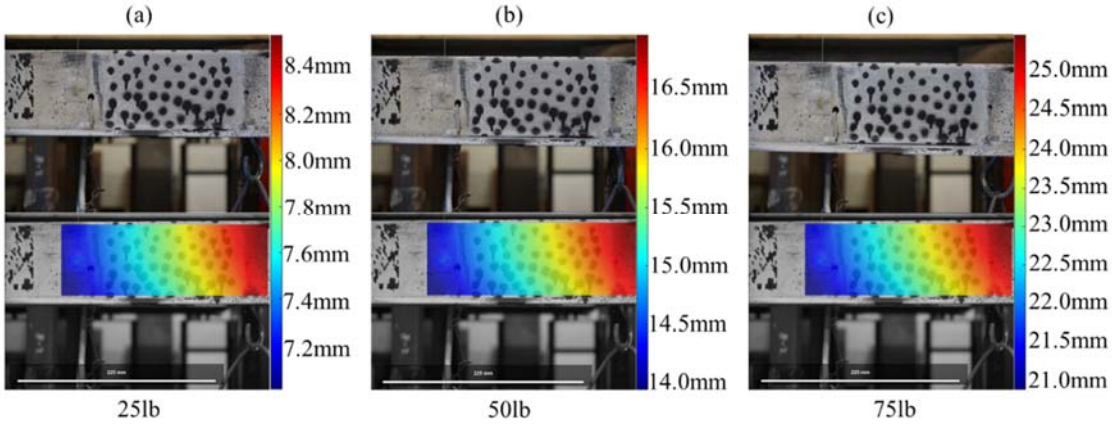


Figure 126. Results of DIC analysis for (a) 25lb, (b) 50lb, and (c) 75lb loads.

After the static experiment, a UAS was used to obtain the same measurements. Since the flight occurred indoors, position control using GPS was not possible, so an optical flow camera and an ultrasonic sensor were placed under the UAS to control the position of the aircraft. Figure 127 (a) shows the schematic of the UAS where the power systems of the aircraft are highlighted in red, the flight control systems are highlighted in blur, and the textured surface under the optical flow camera is highlighted in yellow. The battery provides power to all of the systems on the UAS. The flight controller receives control inputs and feedback from the receiver, optical flow camera, and the ultrasonic sensor and sends the command to the electronic speed controller using this feedback and feedback

from the internal measurement unit to keep the system stable. A textured surface was placed on the floor to ensure a good position hold could be achieved because optical flow requires texture to be effective. The floor by itself was did not provide the system with enough points to allow the UAS to achieve and accurate position hold. Figure 127 (b) shows the UAS flying over the textured surface during the experiment.

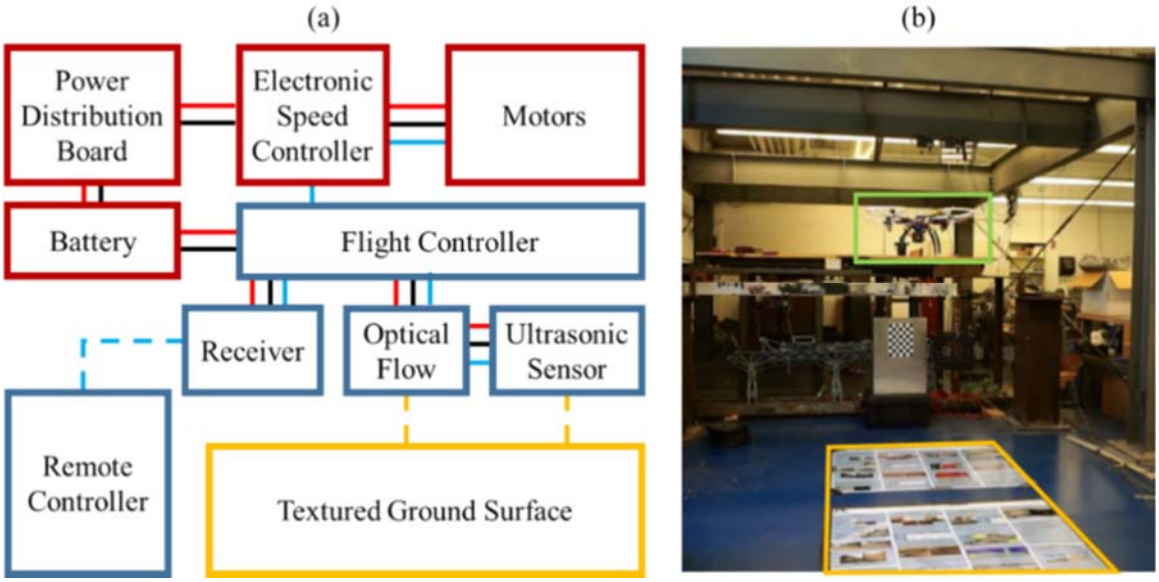


Figure 127. (a) Diagram of UAV power (red and black), control (blue), and sensing (yellow). (b)Picture of the UAV (green) flying over the textured surface (yellow).

Due to the lift limitations of the UAS, a gimbal was not able to be added to the system, so a damping system was added and the optical flow camera was relied upon to keep the system still enough to take clear imagery without a gimbal. After the test, the images were preprocessed before loading them into Ncorr. The steps for processing are shown in Figure 128 (a). The results of the preprocessing steps are shown in Figure 128 (b), (c), and (d).

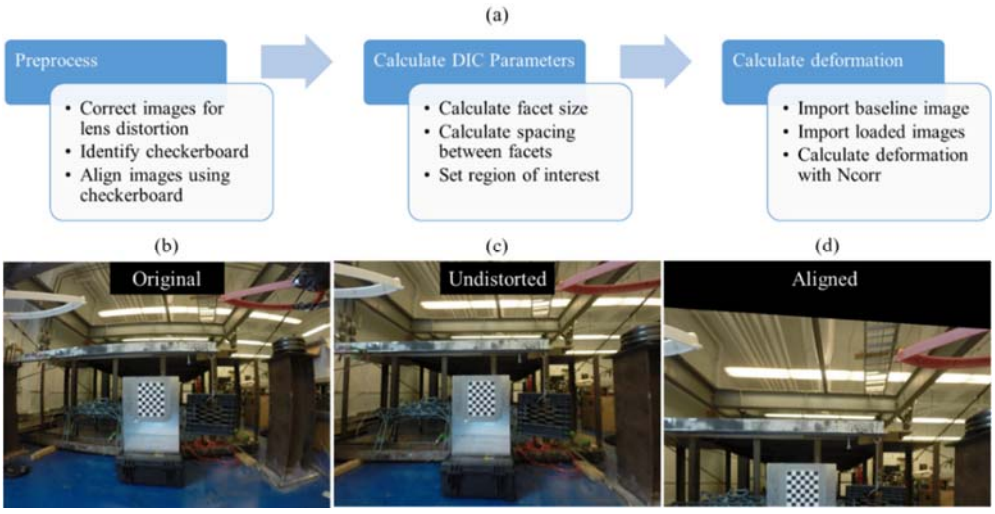


Figure 128. (a) Algorithm for UAV analysis. (b) Image obtained from the UAV. (c) Image corrected for lens distortion. (d) Image aligned using the checkerboard.

The original image was first corrected for lens distortion, then the checkerboard was then identified, the corners in the checkerboard were extracted using edge detection and Hough transform and the homography matrix was calculated. Finally, the images were projected onto the defined by the checkerboard. This forced all of the images taken during the flight to be in the same plane to ensure they were directly comparable. After preprocessing, the aligned images were then imported into Ncorr. The 2D DIC results for 25lb, 50lb, and 75lb are shown in Figure 129.

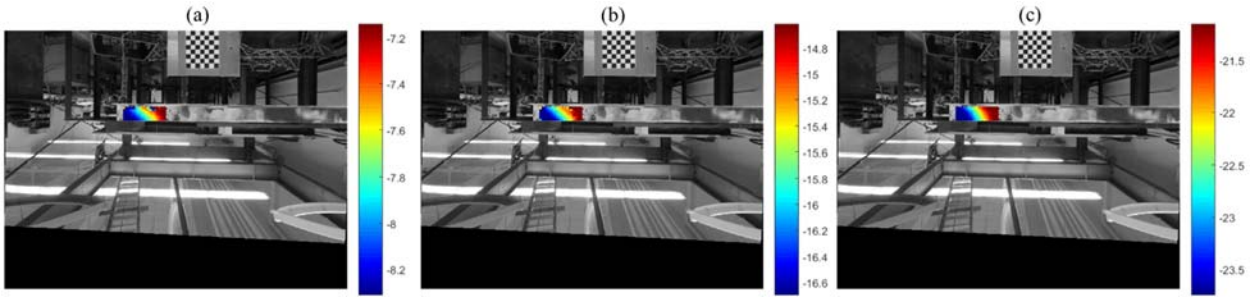


Figure 129. Output results from Ncorr with colorbar corresponding to the displacement for (a) 25lbs, (b) 50lbs, and (c) 75lbs.

The image based results from the DIC software for both the high resolution static and hovering UAS were compared to the displacement gauges using the pixels nearest to the gauge. The results of all of the measurement systems were compared to the theoretical displacements obtained using Euler-Bernoulli beam theory and the vertical displacement obtained from the finite element model. The results of each measurement system and the theoretical calculations are shown below in Table 24 and the error with respect to the displacement gauges are shown in Table 25.

Table 24. Results from static imagery, UAV imagery, displacement gauges, the Euler-Bernoulli beam theory and ABAQUS results.

Load (lb)	Static (mm)	UAV (mm)	Gauge (mm)	Euler-Bernoulli (mm)	ABAQUS (mm)
0	0	0	0	0	0
25	-7.00	-7.21	-7.01	-6.91	-7.33
50	-14.31	-14.81	-14.23	-13.81	-14.67
75	-21.82	-21.19	-21.59	-20.72	-22.00

Table 25. Percent error from static imagery, UAV imagery, the Euler-Bernoulli beam theory and ABAQUS results.

Load (lb)	Static (%)	UAV (%)	Gauge (%)	Euler-Bernoulli (%)	ABAQUS (%)
0	-	-	-	-	-
25	0.14	-2.85	-	1.43	-4.56
50	-0.56	-4.08	-	2.95	-3.09
75	-1.07	1.85	-	4.03	-1.9

The results in Table 24 and Table 25 assume there is no out of plane motion and as a result only the vertical motion is displayed. The beam was asymmetric and therefore there was out of plane motion which could significantly affect the results of the experiment if the beam was loaded with enough weight. The Euler-Bernoulli beam used to design the

experiment did not take this into account and the vertical direction movement is smaller than that of the displacement gauges and the DIC measurements in all cases suggesting that the out of plane motion did affect the measurements slightly. The out of plane motion forced the beam away from the camera which should make the image measurements slightly smaller than expected.

The working distance for the static camera is much smaller than the UAS camera working distance. The pixel size in the images taken with the static camera was 0.05mm/pixel, which is significantly much higher quality than the 0.68mm/pixel associated with UAV imagery. This difference in pixel size was caused by the different camera resolutions, different lenses, and different working distances. The movement corrections, blur caused by vibrations, and the residual errors from the lens distortion correction added even more error to the UAS measurements. Adding a vibration dampening gimbal and higher resolution camera with increased zoom capabilities would improve the UAS DIC results significantly.

Chapter 7 Concluding Remarks

7.1 Summary of Contributions

This thesis presented the feasibility of using an Unmanned Aerials System (UAS) for Structural Health Monitoring (SHM) applications focusing on the identification of potentially damaged areas and measurement with remote sensing systems. Though bridges were used as a representative example, the work presented is not limited to bridge applications, but can easily extend to many types of infrastructure such as buildings, dams, and roadways. Most existing work using UAS for inspection has been limited to mapping and taking images of a structure leaving it up to the inspection personnel to identify and quantify damage from the images to make decisions [299, 300]. This is an important first step to obtaining useful information, however in general, owners of infrastructure systems need to know if they need to perform maintenance, stop using the structure, or perform a detailed inspection and are not interested in having more data to search through.

Many image processing techniques have been developed for identification of damage with varying degrees of success [40, 212, 230]. Though many methods have been proposed, there is still no segmentation technique that will work on all image sets. Furthermore, the methods for crack identification are simplified due to the fact that only the structure is visible in the image and in general, the object in the image is planar [40, 130]. The method proposed in Chapter 4 of this thesis were used to identify cracks from UAS imagery for a lab setting and focus on both the identification of cracks as well as the elimination of the parts of the image that cannot have cracks. Most methods used for image based crack identification only use images that have only the areas in question in the image.

In UAS imagery, this is almost never the case because there will always be something other than the structure visible in UAS imagery. The global visualization using Structure From Motion (SFM) for simple reconstruction is not new, however the damage overlay on the structure is a major advantage to the method proposed in Chapter 4 and Chapter 5. The global damage map is an acceptable output that can be presented to an owner or other decision maker to directly compare structures to previous inspections or other structures. This is crucial to asset management because the distribution of funding is difficult when there is no direct comparison available between structures [301]. Furthermore, machine learning techniques can be implemented to ensure that the objects identified in the image are damage features and not other artifacts within the image.

In Chapter 6, several methods for image based measurements were presented for use with and without UAS. The point based methods produce similar results to contact sensors, but without the hassle of running cables, requiring a ground reference for each target, and without problems associated with environmental conditions such as wind which causes a large amount of noise in the measurements. The point based image measurements have their own set of limitations also described in Chapter 6, but do present an alternative to contact sensors. Full field Digital Image Correlation (DIC) was also presented in Chapter 6 and novel multiscale multispectral method for obtaining measurements was presented which would be beneficial to use on UAS because it allows for multiple speckle patterns optimized for different working distances. This is crucial to use on UAS because they generally do not maintain a constant distance from the structure. In addition, most data that is acquired will not have any useful information, so a multiscale method was proposed to capture images at a farther working distance and if something unusual appears in those

measurements, then a image set could be leveraged to obtain higher detail in the areas in question. The use of UAS in conjunction with a DIC speckle pattern was also presented in Chapter 6. A ground reference was required for measurements, but the location of the reference was not required to be directly under the sensor and the only requirement was that the reference did not move. Furthermore, the UAS full field measurements provide more information to characterize local deformation and, in the future, strains between the contact sensors. This local information is crucial in the cases where local damage is present, for example cracking or areas of higher strain.

The bridge technology integration matrix presented in Figure 6 is redrawn again in Figure 130 with the contributions highlighted in blue with gold boarder. The light blue sections are areas in which UAS could be used in the future. The light gray boxes show areas in which other NonDestructive Testing (NDT) methods, engineering expertise, or other methods are required to progress farther in these areas. The in-depth visual inspection using UAS and capturing noncontact geometry were presented in Chapter 4 and Chapter 5. The full views of the structure would contribute to developing historical data for a bridge information model (BRIM) and contribute to the information warehouse. Chapter 4 and Chapter 5 also present the ability of UAS to collect color and IR imagery to perform wide-area NDE scans. The measurement methods presented in Chapter 6 could contribute to controlled load testing and load rating. This, in conjunction with the existing contact sensors already implemented in the field, can provide a greater degree of detail for the measurements. All of the techniques implemented through time contribute to the SHM framework. Combining many of these techniques with computer vision and machine

learning will also help automate some of the processes involved in leveraging the technology for civil structures.

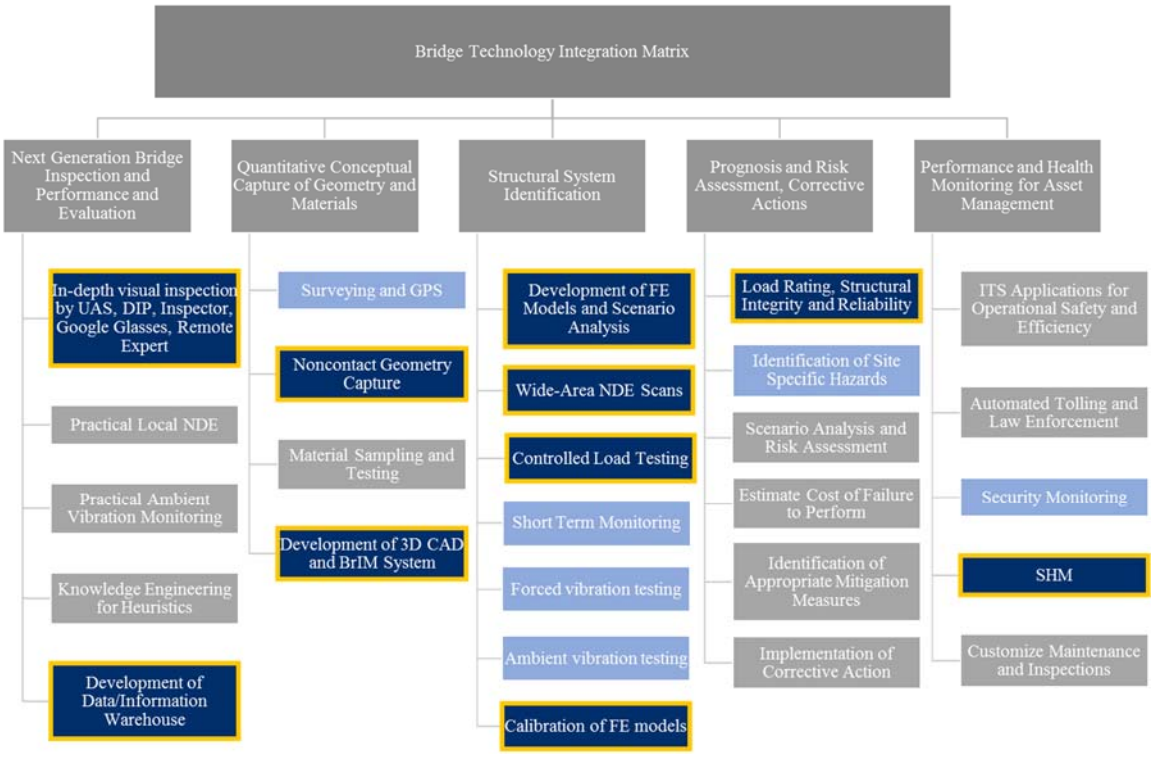


Figure 130. Bridge technology integration matrix highlighting the contributions of the research presented in this thesis

7.2 Future Direction

The future of SHM and the integration of technology into civil infrastructure assessment will revolutionize the industry. Sensor networks integrated into the structure and systems that provide notifications of damage and changes in the structure to the decision makers. This has started to take effect in other countries such as the Rion Antirion bridge in Greece which has several sensors permanently installed which can be used to send notifications to decision makers when specific events occur [15]. Integrating sensors

into the structure is just the beginning. If DIC speckle patterns are manufactured onto the structural elements, there will be no need to apply the pattern after the structure is in place. Furthermore, invisible paints, printed materials, or other advanced manufacturing method, that do not emit light into the visible spectrum could be used to implement this pattern and not degrade the aesthetic appearance of the structure [302]. If new infrastructure is built with SHM in mind, significant improvements can be made in monitoring the structure and it will reduce costs of integrating the system with SHM equipment.

The integration of UAS into the SHM framework will have a massive impact on infrastructure assessment. The control systems on UAS could eventually not rely on GPS and instead use image based systems such as a VICON system to obtain high accuracy position information on the UAS [303]. This would be particularly useful in locations where GPS performance is degraded in places like under bridges or near buildings. Furthermore, multiple UAS can be used in a swarm to capture data on a structure to increase speed or perform different data capture tasks [304]. High accuracy position information can be leveraged to perform quantitative measurements of global movements such as settling with systems like LiDAR. 3D Simultaneous Localization And Mapping (3D SLAM) has been used for mapping of unknown environments and determining position based on the map [117]. This could also be leveraged in the control of systems around structures to ensure collisions do not occur, which is another big challenge to flying UAS around structures in field conditions. Wind is a major factor in environmental conditions that will heavily effect the performance of a UAS flying around a structure. In addition, the inaccurate position information makes it difficult to localize the UAS with respect to the structure making collisions possible. High accuracy position information of

both the UAS and the structure location built into the control of the system would make it possible to safely fly around a structure to collect data. Furthermore, additional sensing systems could be implemented on UAS in the future when the weights of the sensors decrease and the flight time of the systems increase. One advantage to having more than on camera on the system is the ability to perform 3D measurements using stereovision [246, 247]. Multispectral and hyperspectral cameras would also be beneficial to UAS based sensing. LiDAR could also be added to obtain geometry information of the structure and in the future could be used to construct more accurate computational models [38, 91].

In general, the sensors and data are not important to the owner of the structure. Owners only need to know if the structure is safe for operation. Therefore, the identification and localization of damage is one of the most important parts of SHM. If the system cannot provide information to the owner, the SHM framework will not be effective. With this in mind, future segmentation algorithms and application of machine learning [33] will greatly advance the field of SHM. Integration of other technologies and multiple sets of data also will aid in SHM and asset management. For instance, integrating the damage information into a GIS database and matching a structure with a digital model that leverages all of the data obtained from previous inspections and building plans would help organize the data in a more efficient way. Creating a computational model with smaller errors would also allow for the interrogation of the model to determine if and where maintenance needs to be performed based on the results of a simulation. In addition, incorporating both contact and noncontact sensor data could improve upon the remote sensing measurements and simplify the acquisition of data on the structure. In some cases, it may eliminate the need for a ground based reference in the image data or an exact camera location. If the reference is

placed on the structure and the displacements on the structure are known at specific points from contact sensors, the contact sensors can account for the motion of the structure at those points and the image measurements can be used for the local measurements to provide full field data. Furthermore, future remote sensing systems will become lighter and have higher resolution which could allow UAS data to calculate strain in structural members in the absence of a ground reference.

The future of SHM will include many different sensing systems integrated into the structure as well as robotic platforms to acquire data, analyze data, and repair the structure. As the technology develops, SHM will be more trusted among the owners of structures and implemented more often than it is today. UAS could be the first response for monitoring with remote sensing, but could also potentially be used for contact sensing as well [95, 305]. The data could then be used to point other robotic systems to areas of interest for a more detailed inspection and repair the structure [20, 306, 307]. A potential framework for future SHM applications is shown in Figure 131. The methodology would start with the collection of data with contact and noncontact sensing systems. The data would be processed to identify damage and measure deformation. A model would then be interrogated to determine if the damage will have an impact on the structure as well as determine if the measurements indicate a significant loss in performance from previous analyses. Then, if necessary, a detailed assessment could be performed with a robotic system or inspection personnel and repairs could be completed leveraging robotic platforms. The process would then cycle back to the collection of data and due to the automated nature of most of the steps, these analyses could be performed every day based on embedded sensors and a UAS programmed to perform a scan at predetermined times.

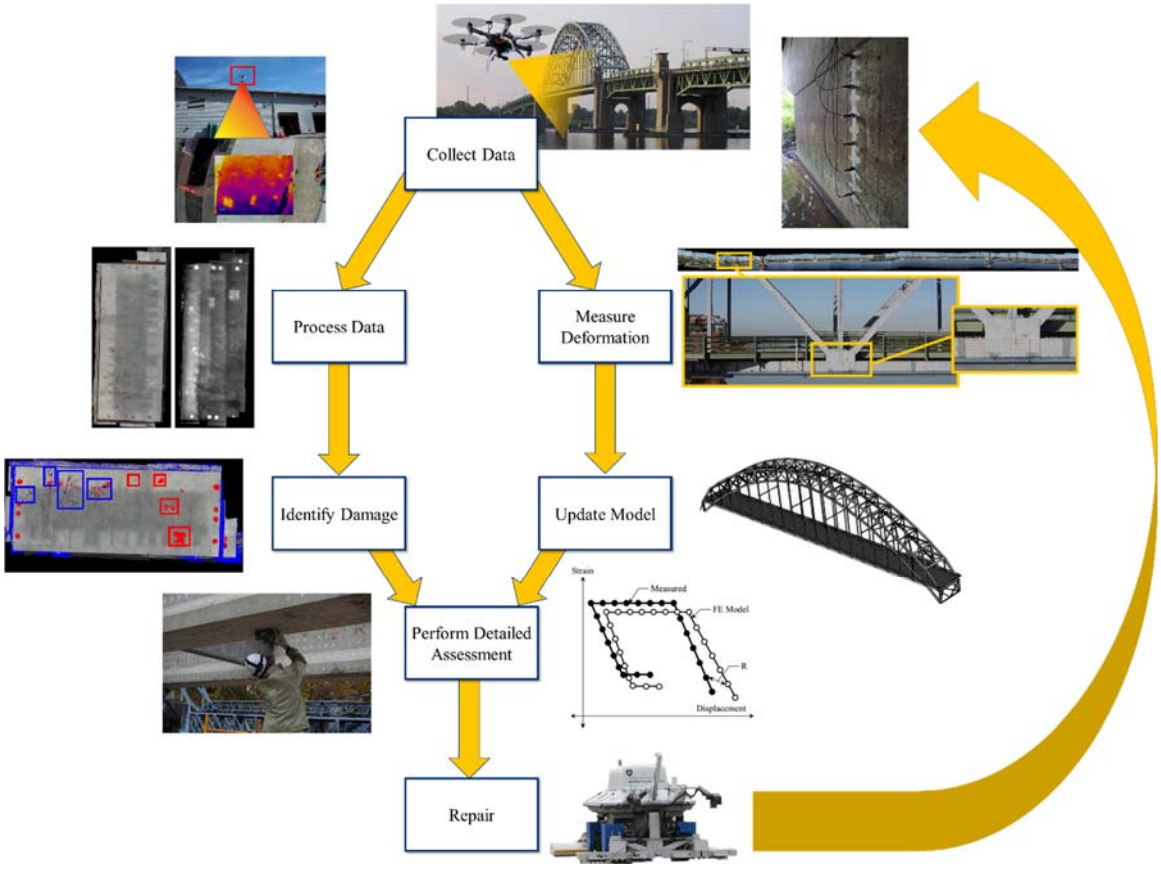


Figure 131. Future vision for SHM

7.3 Future Work

7.3.1 Model Updating

Updating an existing finite element model could be a major benefit to SHM in the future to allow testing of potential scenarios to determine how the structure will perform. One potential method that could be used to complete this is full field DIC measurements used to create boundary conditions in the model. Preliminary steps towards this task were completed and presented in this section. The expected steps to complete this task leveraging full field DIC, LiDAR, and/or SFM are shown in Figure 132. The first step is

to register the experimental data with a well-known computational model of the structure. Then, the boundary conditions would be set based on the experimental measurements of deformation. The model was created in ABAQUS and registered with DIC measurements calculated in ARAMIS using MATLAB [120, 288, 308]. After registration, the measurements were interpolated to match the locations of the finite element model. Finally, the points were compared to determine how different the experimental data is from the theoretical analysis in the finite element model.

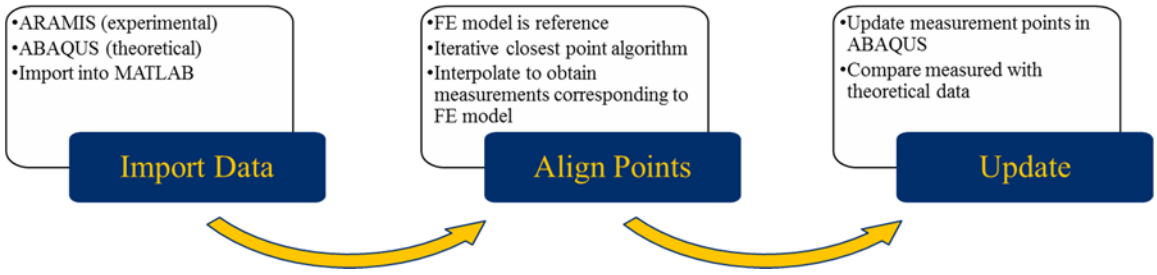


Figure 132. Data process for FEM update with experimental data

7.3.2 Preliminary Results

The model from section 6.4.1 was used along with the displacement measurements obtained from the experiment at the close distance. The iterative closest point algorithm was used to match the locations of the point cloud output from both ABAQUS and ARAMIS [44, 45]. Since the motion of the end of the beam is close to rigid body, the assumption is not violated and the initial transformation was completed using the manual measurement of the translation between the two point clouds. Figure 133 shows the misalignment of the point clouds and highlights the different densities of points shown in (c) and (d).

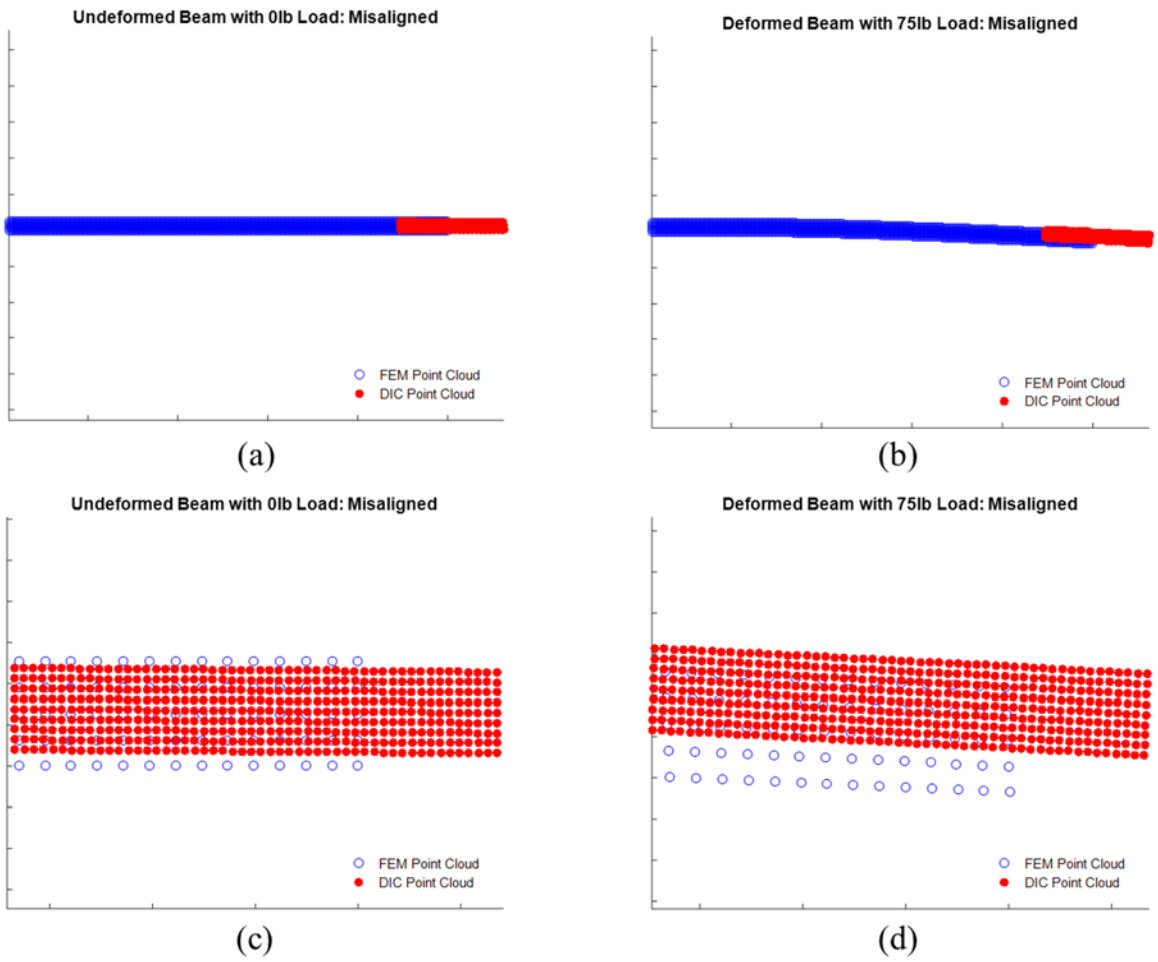


Figure 133. Misaligned point clouds are shown in (a) and (b). Zoomed in versions are shown in (c) and (d).

The undeformed model and the undeformed experimental data were used to align the model and the experimental results for all cases since the camera did not move during the test. This made it possible to align the data without errors associated with the mismatch in deformation between the model and the experiment. After alignment, the points are overlaid on top of each other as shown in Figure 134. It is clear in (c) and (d) that the different point densities did not make a difference in the alignment.

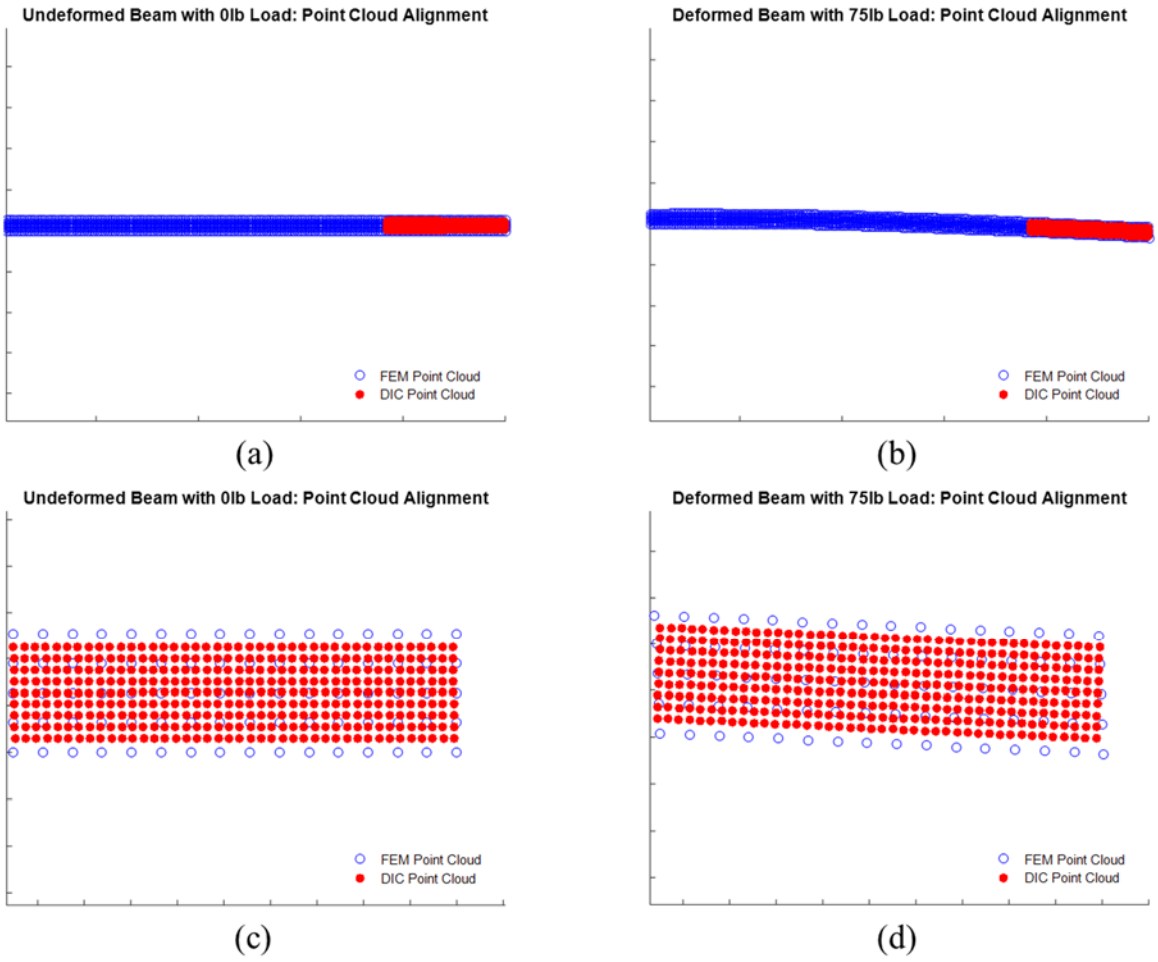


Figure 134. Aligned point clouds are shown in (a) and (b). Zoomed in versions are shown in (c) and (d).

The DIC point cloud is only in two dimensions which is demonstrated in Figure 133 and Figure 134 where the FEM point cloud is in 2D with only the points corresponding to the points on the front face of the C-beam. The experimental displacements calculated in section 6.4.1 were interpolated to match the locations of the points on the FEM point cloud. Different types of interpolation can be used with different assumptions about the errors. Linear interpolation was used to demonstrate the concept, however higher order polynomials generally achieve better results. The DIC measurements were interpolated onto the FEM point cloud to ensure that the points aligned properly with the nodes of the

finite element model. The result of this process is shown in Figure 135. The red x's are the nodes of the finite element simulation under 75lb load and the blue circles are the DIC measurements from ARAMIS interpolated onto the finite element nodes.

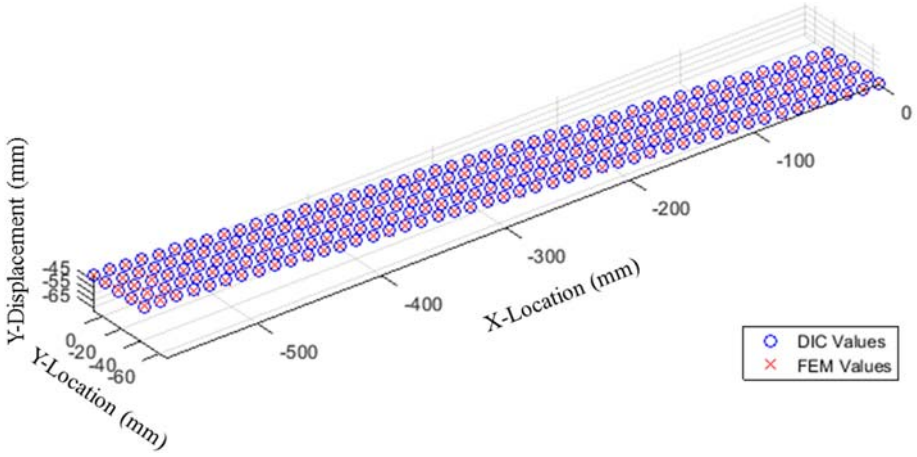


Figure 135. Aligned and interpolated DIC measurements with finite element model at 75lb load

Each node was then set as a boundary condition in the finite element model. The boundary conditions were written to a text file using MATLAB and imported into ABAQUS. Since the measurements were two dimensional, the first attempt at fixing the boundary conditions used a two dimensional model. This neglects several important parameters such as deformation in the third dimension and cross section of the beam which result in a stiffer cross section. The text file written by MATLAB was used to enforce the boundary conditions. The model had 20,181 nodes, 19,200 CPS4R elements, and over 4,500 boundary conditions obtained from the DIC measurements. The color imagery at 2m was used at 25lb and 50lb experimental load to produce the images shown in Figure 136 where the top image imported the 50lb load boundary conditions and the bottom image imported the 25lb load results and then the 50lb results.

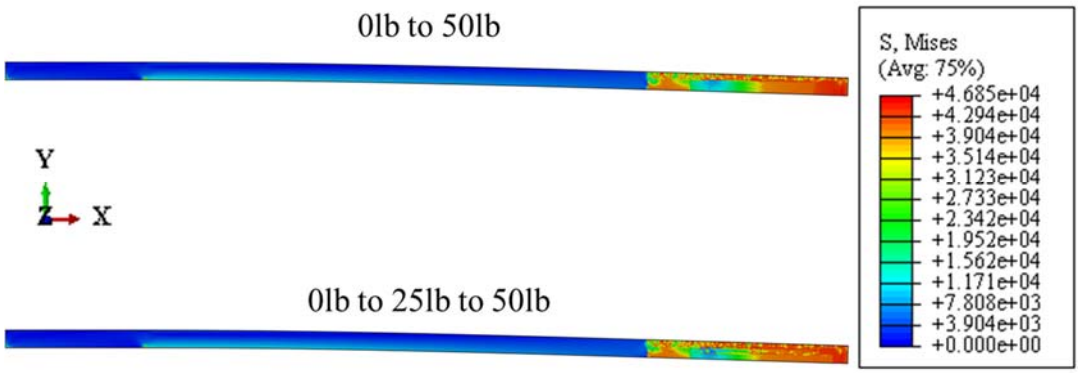


Figure 136. Stress comparison for different load paths from 0lb to 50lb (top) and 0lb to 25lb to 50lb (bottom)

Since the beam was three dimensional and symmetric and the measurements were taken of a two dimensional surface, the same displacements calculated on the front face were applied to the back face in the model. The result of the simulation is shown in Figure 137 leveraging the base model shown in Figure 117. The displacements are shown in inches.

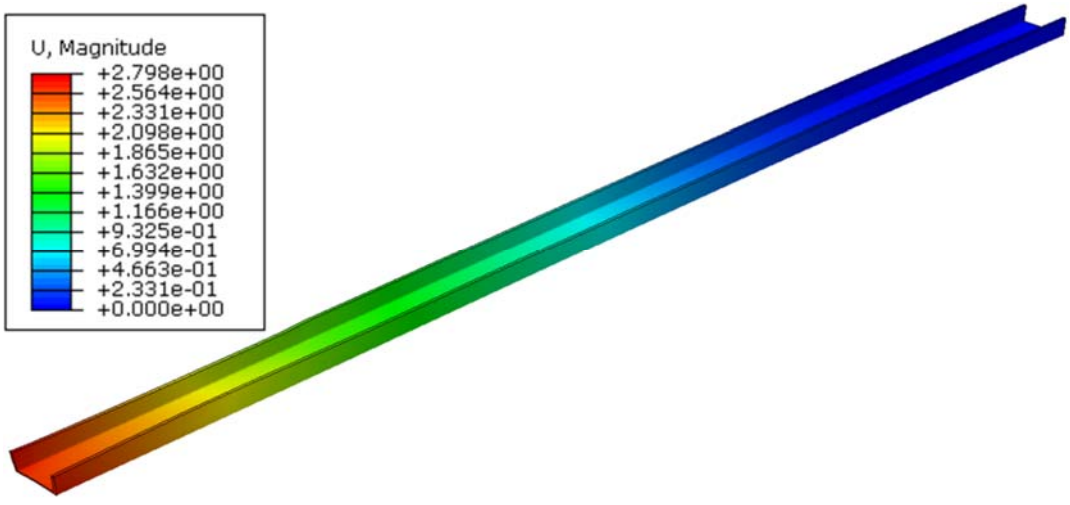


Figure 137. Magnitude of displacements for imposed boundary conditions associated with DIC measurements in inches.

As expected, the DIC displacement results are almost identical to the model so the plots are almost identical as well, however when the stress distribution is viewed in the area of interest where the boundary conditions were applied, the results are clearly different as shown in Figure 138. In the theoretical model, the stress distribution is more uniform with no stress concentrations. The cause of these stress concentrations are the errors in the DIC results due to the large size and lack of uniformity of the speckles within the patterns.

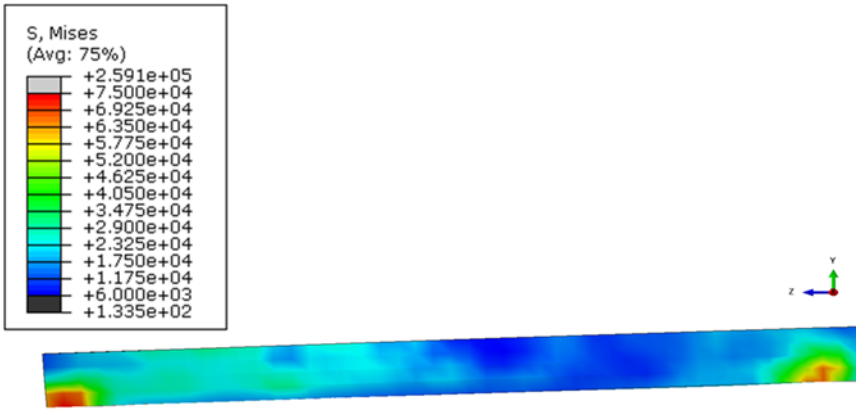


Figure 138. Von Mises stress distribution (psi) at the end of the beam where the boundary conditions were applied.

After deforming the beam, further analysis can performed to determine how much additional load can be applied to the structure before damage would occur such as yielding. The geometry and stress distribution were exported and imported into a new model and the imported deformation and stress distribution are shown in Figure 139 (a) and (b) respectively. This model was then interrogated using higher loading until the beam was yielded. Though this is a simplified model, it does demonstrate the goal for the updating of an existing model with full field experimental data.

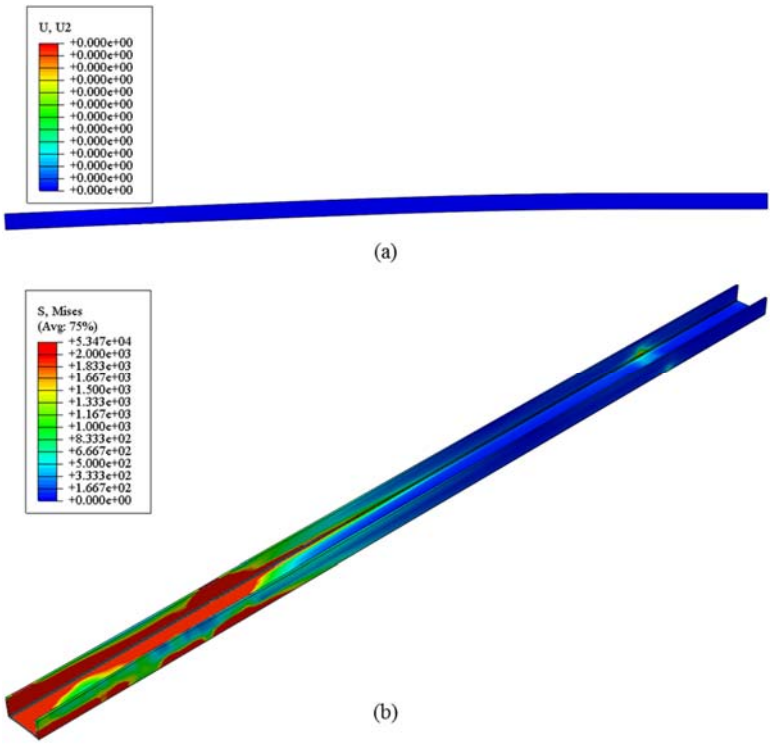


Figure 139. (a) Deformed shape of the beam set as the initial step. (b) Stress distribution imposed on the initial step.

The same methods were extended to UAS measurements using the image data from section 6.4.2 where the displacement measurements were calculated in ARAMIS. The model shown in Figure 125 was used to impose the displacement and the result of the imposed deformation is shown in Figure 140.

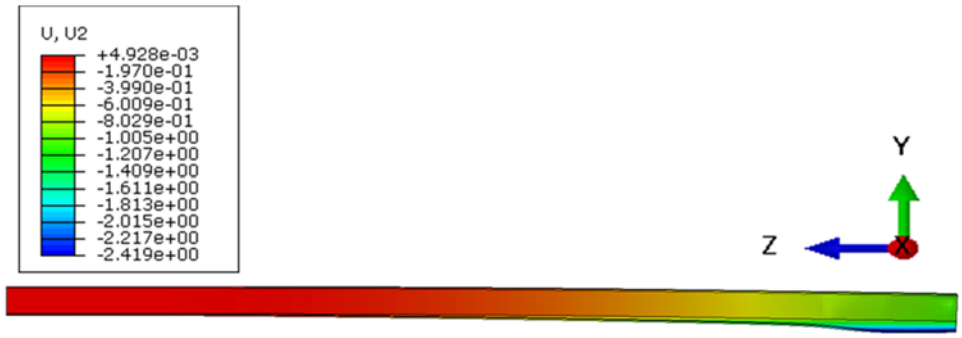


Figure 140. Displacement results in the y-direction in inches

Since the measurement did not take into account the out of plane motion, the results at the end of the beam appear to be more distorted than expected. This combined with interpolation errors and imperfect loading and boundary conditions creates a significant amount of displacement that does not appear in the experimental setup. More work must be done to demonstrate the value of this information for more complicated structures, but in the future, 3D DIC and higher accuracy measurements from aerial systems could have a large impact on infrastructure assessment [246, 247]. In addition, better DIC results leveraging smaller speckle patterns will decrease the error which will limit the stress concentration in the updated model. After enforcing the boundary conditions in the model, the model stiffness could be updated and the model could then be interrogated using the current state of the structure. Though the results presented are limited and require much work to become useful, finite element updating leveraging UAS data is possible in the future and can add to the benefit of UAS to SHM applications in the future.

7.4 Extensions

The work in this thesis is not limited to bridges, though it was the type of structure that was emphasized in this thesis. The shortest extension of this work is to other types of infrastructure including buildings [86, 87], pipelines [88], powerlines [89], and many other forms of infrastructure. In general, the damage types are similar for infrastructure because the materials are similar and therefore the identification algorithms should also be quite similar. The challenges associated with different types of infrastructure are expected to vary slightly and therefore the SHM strategy may incorporate more or different types of sensors, but large structures that are difficult and time consuming to assess can benefit from the use of UAS as a data acquisition platform.

Another research area that can benefit from UAS remote sensing is crop monitoring and vegetation health tracking [81, 85, 221]. Healthy plants emit light at a different wavelength from unhealthy plants. This allows infrared or near infrared imagery to distinguish between the two types of plants. The identification algorithms presented in this thesis would need to be adapted to identify the unhealthy plants, but this is not a far reach either. Furthermore, green infrastructure such as rain gardens and green roofs could benefit from this work as well. These systems rely heavily on plants to clean and retain storm water so other water resources do not get overwhelmed [309]. Since they filter out a lot of chemicals, the health of the plants can be degraded which decreases the effectiveness of the systems. The same methods with infrared imagery could be applied to these systems as well to help determine if the plants are healthy, if the systems are retaining too much water, or if there is something interfering with the performance of the system [309, 310].

This work could also extend to the analysis of ground or aerial vehicles for damage identification and measurement using a multispectral speckle pattern. For instance, at airports when an aircraft lands, a scan could be completed to determine if any cracks had propagated during the flight and any red flags could be used to target a more detailed inspection and analysis to determine if the aircraft was ready to perform another flight. These methods in conjunction with other embedded sensors onboard during flight could move inspections from a time based assessment to a condition based assessment [311].

References

[1] P. A. Dalton, *Physical Infrastructure: Challenges and Investment Options for the Nation's Infrastructure: Testimony Before the Committee on the Budget and the Committee on Transportation and Infastructure, US House of Representatives*: US Government Accountability Office, 2008.

[2] ASCE. (2013, 9/6). *2013 ASCE's Report Card for America's Infrastructure*. Available: http://2013.infrastructurereportcard.org/

[3] EIA. (2012). *2012 CBECS Survey Data*. Available: https://www.eia.gov/consumption/commercial/data/2012/#b6-b10

[4] ASCE. (2017). *2017 ASCE's Report Card for America's Infrastructure*. Available: http://www.infrastructurereportcard.org/

[5] A. Aktan, F. Moon, I. Bartoli, and K. Sjoblom, "Identification of Infrastructure Systems," *Journal of Infrastructure Systems,* p. 02516002, 2016.

[6] FHWA. (2010). *Conditions and Performance Chapter 3 System Conditions*. Available: https://www.fhwa.dot.gov/policy/2010cpr/chap3.cfm#8

[7] FHWA. (2015, 9/6). *Deficient Bridges by Highway System 2015*. Available: http://www.fhwa.dot.gov/bridge/deficient.cfm

[8] B. M. Phares, G. A. Washer, D. D. Rolander, B. A. Graybeal, and M. Moore, "Routine highway bridge inspection condition documentation accuracy and reliability," *Journal of Bridge Engineering,* vol. 9, pp. 403-413, 2004.

[9] B. A. Graybeal, B. M. Phares, D. D. Rolander, M. Moore, and G. Washer, "Visual inspection of highway bridges," *Journal of nondestructive evaluation,* vol. 21, pp. 67-83, 2002.

[10] K. Vaghefi, R. C. Oats, D. K. Harris, T. M. Ahlborn, C. N. Brooks, K. A. Endsley, *et al.*, "Evaluation of commercially available remote sensors for highway bridge condition assessment," *Journal of Bridge Engineering,* vol. 17, pp. 886-895, 2011.

[11] N. Metni and T. Hamel, "A UAV for bridge inspection: Visual servoing control law with orientation limits," *Automation in construction,* vol. 17, pp. 3-10, 2007.

[12] P. A. Vanniamparambil, R. Carmi, F. Khan, J. Cuadra, I. Bartoli, and A. Kontsos, "An active–passive acoustics approach for bond-line condition monitoring in aerospace skin stiffener panels," *Aerospace Science and Technology,* vol. 43, pp. 289-300, 2015.

[13] F. Khan, S. Rajaram, P. A. Vanniamparambil, M. Bolhassani, A. Hamid, A. Kontsos*, et al.*, "Multi-sensing NDT for damage assessment of concrete masonry walls," *Structural Control and Health Monitoring,* 2014.

[14] A. Ellenberg, A. Kontsos, F. Moon, and I. Bartoli, "Low-cost, quantitative assessment of highway bridges through the use of unmanned aerial vehicles," in *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, 2016, pp. 98052D-98052D-10.

[15] A. Panagis, "The structural health monitoring system of the Rion Antirion Bridge “Charilaos Trikoupis”," 2016.

[16] R. Maaskant, T. Alavie, R. Measures, G. Tadros, S. Rizkalla, and A. Guha-Thakurta, "Fiber-optic Bragg grating sensors for bridge monitoring," *Cement and Concrete Composites,* vol. 19, pp. 21-33, 1997.

[17] K. L. Rens, T. J. Wipf, and F. W. Klaiber, "Review of nondestructive evaluation techniques of civil infrastructure," *Journal of performance of constructed facilities,* vol. 11, pp. 152-160, 1997.

[18] P. A. Vanniamparambil, M. Bolhassani, R. Carmi, F. Khan, I. Bartoli, F. L. Moon*, et al.*, "A data fusion approach for progressive damage quantification in reinforced concrete masonry walls," *Smart Materials and Structures,* vol. 23, p. 015007, 2013.

[19] F. Khan and I. Bartoli, "Detection of delamination in concrete slabs combining infrared thermography and impact echo techniques: A comparative experimental study," in *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, 2015, pp. 94370I-94370I-11.

[20] N. Gucunski, S. Kee, H. La, B. Basily, A. Maher, and H. Ghasemi, "Implementation of a Fully Autonomous Platform for Assessment of Concrete Bridge Decks RABIT," in *Structures Congress 2015*, 2015.

[21] T. Oh, S.-H. Kee, R. W. Arndt, J. S. Popovics, and J. Zhu, "Comparison of NDT methods for assessment of a concrete bridge deck," *Journal of Engineering Mechanics,* vol. 139, pp. 305-314, 2012.

[22] L. Shoup, N. Donohue, M. Lang, T. Mejia, S. Barry, D. Goldberg*, et al.*, "The Fix We’re In For: The State Of Our Nation’s Bridges," 2011.

[23] C. R. Farrar and K. Worden, "An introduction to structural health monitoring," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences,* vol. 365, pp. 303-315, 2007-02-15 00:00:00 2007.

[24] N. T. S. Board, "Collapse of I-35W Highway Bridge Minneapolis, Minnesota August 1, 2007. 2008," *National Transportation Safety Board: Washington DC,* p. 1.

[25] J. Zink and B. Lovelace, "Unmanned aerial vehicle bridge inspection demonstration project," 2015.

[26] D. F. Laefer, J. Gannon, and E. Deely, "Reliability of Crack Detection Methods for Baseline Condition Assessments," *Journal of Infrastructure Systems,* vol. 16, pp. 129-137, Jun 2010.

[27] M. R. Jahanshahi, S. F. Masri, C. W. Padgett, and G. S. Sukhatme, "An innovative methodology for detection and quantification of cracks through incorporation of depth perception," *Machine Vision and Applications,* vol. 24, pp. 227-241, Feb 2013.

[28] A. Khaloo and D. Lattanzi, "Hierarchical Dense Structure-from-Motion Reconstructions for Infrastructure Condition Assessment," *Journal of Computing in Civil Engineering,* p. 04016047, 2016.

[29] D. Lattanzi and A. Khaloo, "Extracting Structural Models through Computer Vision," 2015.

[30] N. Hallermann and G. Morgenthal, "Visual inspection strategies for large bridges using Unmanned Aerial Vehicles (UAV)," in *7th International Conference on Bridge Maintenance, Safety and Management, IABMAS 2014, July 7, 2014-July 11*, 2014.

[31] H. Sohn, C. R. Farrar, F. M. Hemez, and J. J. Czarnecki, "A Review of Structural Health Review of Structural Health Monitoring Literature 1996-2001," Los Alamos National Laboratory2002.

[32] K. Worden, C. R. Farrar, G. Manson, and G. Park, "The fundamental axioms of structural health monitoring," in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 2007, pp. 1639-1664.

[33] K. Worden and G. Manson, "The application of machine learning to structural health monitoring," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences,* vol. 365, pp. 515-537, 2007.

[34] F. N. Catbas, S. K. Ciloglu, O. Hasancebi, K. Grimmelsman, and A. E. Aktan, "Limitations in structural identification of large constructed structures," *Journal of Structural Engineering,* vol. 133, pp. 1051-1066, 2007.

[35] E. Aktan, "MEM 678 Structural Health Monitoring Lecture," in *Nondestructive Evaluation Methods*, ed, 2015.

[36] B. Guldur, Y. Yan, and J. F. Hajjar, "Condition Assessment of Bridges Using Terrestrial Laser Scanners," in *Structures Congress 2015*, 2015, pp. 355-366.

[37] D. Lattanzi and G. R. Miller, "3D Scene Reconstruction for Robotic Bridge Inspection," *Journal of Infrastructure Systems,* vol. 21, p. 04014041, 2014.

[38] G. Castellazzi, A. M. D'Altri, G. Bitelli, I. Selvaggi, and A. Lambertini, "From laser scanning to finite element analysis of complex buildings by using a semi-automatic procedure," *Sensors,* vol. 15, pp. 18360-18380, 2015.

[39] H. Fathi, F. Dai, and M. Lourakis, "Automated as-built 3D reconstruction of civil infrastructure using computer vision: achievements, opportunities, and challenges," *Advanced Engineering Informatics,* vol. 29, pp. 149-161, 2015.

[40] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure," *Adv. Eng. Inform.,* vol. 29, pp. 196-210, 2015.

[41] K. D. Hjelmstad and S. Shin, "Damage detection and assessment of structures from static response," *Journal of engineering mechanics,* vol. 123, pp. 568-576, 1997.

[42] Q. Mao, J. Devitis, M. Mazzotti, I. Bartoli, F. Moon, K. Sjoblom*, et al.*, "Boundary condition identification for a grid model by experimental and numerical dynamic analysis," in *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, 2015, pp. 94370J-94370J-11.

[43] M. Sanayei, G. R. Imbaro, J. A. McClain, and L. C. Brown, "Structural model updating using experimental static measurements," *Journal of structural engineering,* vol. 123, pp. 792-798, 1997.

[44] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," in *Robotics-DL tentative*, 1992, pp. 586-606.

[45] D. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*: Pearson, 2012.

[46] J. M. Brownjohn, P.-Q. Xia, H. Hao, and Y. Xia, "Civil structure condition assessment by FE model updating:: methodology and case studies," *Finite elements in analysis and design,* vol. 37, pp. 761-775, 2001.

[47] B. Jaishi and W.-X. Ren, "Damage detection by finite element model updating using modal flexibility residual," *Journal of sound and vibration,* vol. 290, pp. 369-387, 2006.

[48] H. Li, S. Dong, S. El-Tawil, and V. R. Kamat, "Relative displacement sensing techniques for postevent structural damage assessment: Review," *Journal of Structural Engineering,* vol. 139, pp. 1421-1434, 2012.

[49] J. L. DeVitis, *Design, Development, and Validation of a Rapid Modal Testing System for the Efficient Structural Identification of Highway Bridges*: Drexel University, 2015.

[50] D. R. Masceri Jr, *Examination of Bridge Performance through the Extension of Simulation Modeling and Structural Identification to Large Populations of Structures*: Drexel University, 2015.

[51] H. Schlune, M. Plos, and K. Gylltoft, "Improved bridge evaluation through finite element model updating using static and dynamic measurements," *Engineering structures,* vol. 31, pp. 1477-1485, 2009.

[52] M. Sanayei, A. Khaloo, M. Gul, and F. N. Catbas, "Automated finite element model updating of a scale bridge model using measured static and modal test data," *Engineering Structures,* vol. 102, pp. 66-79, 2015.

[53] S. Siebert, J. Klonowski, and F. Neitzel, "Unmanned Aerial Vehicles (UAV)– Historische Entwicklung, rechtliche Rahmenbedingungen und Betriebskonzepte."

[54] J. Peraire and S. Widnall, "Lecture L29-3D Rigid Body Dynamics," *Dynamics,* 2009.

[55] NTIA. (2013). *United States Frequency Allocation Chart*. Available: http://www.ntia.doc.gov/page/2011/united-states-frequency-allocation-chart

[56] M. Osborne, "Mission Planner," 1.3.38 ed, 2016.

[57] L. Meier, "QGroundControl," ed, 2017.

[58] F. Kendoul, I. Fantoni, and K. Nonami, "Optic flow-based vision system for autonomous 3D localization and control of small aerial vehicles," *Robotics and Autonomous Systems,* vol. 57, pp. 591-602, 2009.

[59] D. Honegger, L. Meier, P. Tanskanen, and M. Pollefeys, "An open source and open hardware embedded metric optical flow cmos camera for indoor and outdoor applications," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 2013, pp. 1736-1741.

[60] T. Butkiewicz, R. Chang, Z. Wartell, and W. Ribarsky, "Visual analysis for live LIDAR battlefield change detection," in *SPIE Defense and Security Symposium*, 2008, pp. 69830B-69830B-10.

[61] P. Liu, A. Y. Chen, Y.-N. Huang, J.-Y. Han, J.-S. Lai, S.-C. Kang*, et al.*, "A review of rotorcraft Unmanned Aerial Vehicle (UAV) developments and applications in civil engineering," *SMART STRUCTURES AND SYSTEMS,* vol. 13, pp. 1065-1094, 2014.

[62] J. D. Blom, *Unmanned Aerial Systems: a historical perspective* vol. 45: Citeseer, 2010.

[63] K. Kim and J. Davidson, "Unmanned Aircraft Systems Used for Disaster Management," *Transportation Research Record: Journal of the Transportation Research Board,* pp. 83-90, 2015.

[64] J. F. Keane and S. S. Carr, "A brief history of early unmanned aircraft," *Johns Hopkins APL Technical Digest,* vol. 32, pp. 558-571, 2013.

[65] P. Yost and P. Distribution, *Rise of the Drones*: PBS Distribution, 2013.

[66] I. Klotz. (2012, U.S. Opening up Airspace to Use of Drones. *Discovery News*. Available: http://www.nbcnews.com/id/46499162/ns/technology_and_science-science/t/us-opening-airspace-use-drones/#.WNuqpG_ythE

[67] *FAA Modernization and Reform Act of 2012*, 2012.

[68] B. Jansen, "FAA approves first commercial drone over land," in *USA Today*, ed: USA Today, 2014.

[69] A. Duquette, "FAA Selects Unmanned Aircraft Systems Research and Test Sites," F. A. Administration, Ed., ed: FAA, 2013.

[70] NCSL. (2017). *Current Unmanned Aircraft State Law Landscape*. Available: http://www.ncsl.org/research/transportation/current-unmanned-aircraft-state-law-landscape.aspx#2

[71] FAA. (2014). *Section 333*. Available: https://www.faa.gov/uas/beyond_the_basics/section_333/

[72] FAA. (2014). *Aircraft Registry*. Available: https://www.faa.gov/licenses_certificates/aircraft_certification/aircraft_registry/UA/

[73] FAA, "Summary of Small Unmanned Aircraft Rule (PART 107)," F. A. Administration, Ed., ed. Washington DC, 2016.

[74] FAA, "Operation and Certification of Small Unmanned Aircraft Systems Rule," D. o. Transportation, Ed., ed, 2016.

[75] FAA, "FAA Releases Drone Registration Location Data," D. o. Transportation, Ed., ed, 2016.

[76] FAA, "Operation and Certification of Small Unmanned Aircraft Systems," D. o. Transportation, Ed., ed, 2016.

[77] V. More, H. Kumar, S. Kaingade, P. Gaidhani, and N. Gupta, "Visual odometry using optic flow for Unmanned Aerial Vehicles," in *Cognitive Computing and Information Processing (CCIP), 2015 International Conference on*, 2015, pp. 1-6.

[78] M. Oborne, "Mission Planner Software," 1.3.38 ed, 2007.

[79] J. Park, S. Im, K.-H. Lee, and J.-O. Lee, "Vision-based SLAM system for small UAVs in GPS-denied environments," *Journal of Aerospace Engineering,* vol. 25, pp. 519-529, 2011.

[80] M. Matsumoto, "Bridge Deck Scanning by Infrared Thermography," in *NDE/NDT for Highways & Bridges: SMT 2016*, 2016, pp. 91-98.

[81] M. A. Hsieh, S. Saripalli, G. Sukhatme, and V. Kumar, "Toward a Science of Autonomy for Physical Systems: Aerial Earth Science," *arXiv preprint arXiv:1609.05783,* 2016.

[82] B. Majidi and A. Bab-Hadiashar, "Real time aerial natural image interpretation for autonomous ranger drone navigation," in *Digital Image Computing: Techniques and Applications, 2005. DICTA'05. Proceedings 2005*, 2005, pp. 65-65.

[83] J. A. Diaz, D. Pieri, K. Wright, P. Sorensen, R. Kline-Shoder, C. R. Arkin*, et al.*, "Unmanned Aerial Mass Spectrometer Systems for In-Situ Volcanic Plume Analysis," *Journal of The American Society for Mass Spectrometry,* pp. 1-13, 2015.

[84] P. Rudol and P. Doherty, "Human body detection and geolocalization for UAV search and rescue missions using color and thermal imagery," in *Aerospace Conference, 2008 IEEE*, 2008, pp. 1-8.

[85] J. Berni, P. Zarco-Tejada, L. Suárez, V. González-Dugo, and E. Fereres, "Remote sensing of vegetation from UAV platforms using lightweight multispectral and thermal imaging sensors," *Int. Arch. Photogramm. Remote Sens. Spatial Inform. Sci,* vol. 38, p. 6, 2009.

[86] N. Hallermann, G. Morgenthal, and V. Rodehorst, "Vision-based deformation monitoring of large scale structures using Unmanned Aerial Systems," in *IABSE Symposium Report*, 2014, pp. 2852-2859.

[87] S. Sankarasrinivasan, E. Balasubramanian, K. Karthik, U. Chandrasekar, and R. Gupta, "Health Monitoring of Civil Structures with Integrated UAV and Image Processing System," *Procedia Computer Science,* vol. 54, pp. 508-515, // 2015.

[88] D. Hausamann, W. Zirnig, G. Schreier, and P. Strobl, "Monitoring of gas pipelines-a civil UAV application," *Aircraft Engineering and Aerospace Technology,* vol. 77, pp. 352-360, 2005.

[89] L. F. Luque-Vega, B. Castillo-Toledo, A. Loukianov, and L. E. Gonzalez-Jimenez, "Power line inspection via an unmanned aerial system based on the quadrotor helicopter," in *Mediterranean Electrotechnical Conference (MELECON), 2014 17th IEEE*, 2014, pp. 393-397.

[90] M. G. Javier Irizarry, Bruce N. Walker, "Usability assessment of drone technology as safety inspection tools," *Journal of Information Technology in Construction,* vol. 17, pp. 194-212, 2012 2012.

[91] D. Roca, S. Lagüela, L. Díaz-Vilariño, J. Armesto, and P. Arias, "Low-cost aerial unit for outdoor inspection of building façades," *Automation in Construction,* vol. 36, pp. 128-135, 12// 2013.

[92] G. Morgenthal and N. Hallermann, "Quality Assessment of Unmanned Aerial Vehicle (UAV) Based Visual Inspection of Structures," *Advances in Structural Engineering,* vol. 17, pp. 289-302, Mar 2014.

[93] F. Khan, A. Ellenberg, M. Mazzotti, A. Kontsos, F. Moon, A. Pradhan*, et al.*, "Investigation on Bridge Assessment Using Unmanned Aerial Systems," in *Structures Congress 2015*, 2015, pp. 404-413.

[94] D. Mascareñas, E. Flynn, C. Farrar, G. Park, and M. Todd, "A mobile host approach for wireless powering and interrogation of structural health monitoring sensor networks," *Sensors Journal, IEEE,* vol. 9, pp. 1719-1726, 2009.

[95] D. D. Mascareñas, "A device for remotely tap-testing difficult-to-access structures using aerial robots," *Conference 9805: Health Monitoring of Structural and Biological Systems X,* 2016.

[96] FHWA, "Bridge Inspector's Reference Manual. NHI 03-002," U. S. Department and o. Transportation, Eds., ed, 2012.

[97] J. Guerrero and Y. Bestaoui, "UAV Path Planning for Structure Inspection in Windy Environments," *Journal of Intelligent & Robotic Systems,* vol. 69, pp. 297-311, 2013/01/01 2013.

[98] O. Araar and N. Aouf, "Visual Servoing of a Quadrotor UAV for the Tracking of Linear Structured Infrastructures," in *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, 2013, pp. 3310-3315.

[99] S. German, I. Brilakis, and R. DesRoches, "Rapid entropy-based detection and properties measurement of concrete spalling with machine vision for post-earthquake safety assessments," *Advanced Engineering Informatics,* vol. 26, pp. 846-858, 10// 2012.

[100] Z. Zhu, S. German, and I. Brilakis, "Detection of large-scale concrete columns for automated bridge inspection," *Automation in Construction,* vol. 19, pp. 1047-1055, 12// 2010.

[101] J.-K. Oh, G. Jang, S. Oh, J. H. Lee, B.-J. Yi, Y. S. Moon, *et al.*, "Bridge inspection robot system with machine vision," *Automation in Construction,* vol. 18, pp. 929-941, 11// 2009.

[102] Y. S. Su, S. C. Kang, J. R. Chang, and S. H. Hsieh, "Dual-Light Inspection Method for Automatic Pavement Surveys," *Journal of Computing in Civil Engineering,* vol. 27, pp. 534-543, Sep 2013.

[103] M. Malesa, D. Szczepanek, M. Kujawińska, A. Świercz, and P. Kołakowski, "Monitoring of civil engineering structures using Digital Image Correlation technique," *EPJ Web of Conferences,* vol. 6, p. 31014, 2010.

[104] E. Caetano, S. Silva, and J. Bateira, "A Vision System for Vibration Monitoring of Civil Engineering Structures," *Experimental Techniques,* vol. 35, pp. 74-82, Jul-Aug 2011.

[105] S. E. Chen, C. Rice, C. Boyle, and E. Hauser, "Small-Format Aerial Photography for Highway-Bridge Monitoring," *Journal of Performance of Constructed Facilities,* vol. 25, pp. 105-112, Mar-Apr 2011.

[106] C. S. Zhang and A. Elaksher, "An Unmanned Aerial Vehicle-Based Imaging System for 3D Measurement of Unpaved Road Surface Distresses," *Computer-Aided Civil and Infrastructure Engineering,* vol. 27, pp. 118-129, Feb 2012.

[107] H. Son, N. Hwang, C. Kim, and C. Kim, "Rapid and automated determination of rusted surface areas of a steel bridge for robotic maintenance systems," *Automation in Construction,* vol. 42, pp. 13-24, 2014.

[108] L. Comba, P. Gay, J. Primicerio, and D. R. Aimonino, "Vineyard detection from unmanned aerial systems images," *Computers and Electronics in Agriculture,* vol. 114, pp. 78-87, 2015.

[109] C. Eschmann, C. Kuo, C. Kuo, and C. Boller, "Unmanned aircraft systems for remote building inspection and monitoring," in *6th European workshop on structural health monitoring*, 2012.

[110] I. Abdel-Qader, S. Yohali, O. Abudayyeh, and S. Yehia, "Segmentation of thermal images for non-destructive evaluation of bridge decks," *NDT & E International,* vol. 41, pp. 395-405, 2008.

[111] C. Eschmann, C. M. Kuo, C. H. Kuo, and C. Boller, "High-Resolution Multisensor Infrastructure Inspection with Unmanned Aircraft Systems," *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.,* vol. XL-1/W2, pp. 125-129, 2013.

[112] Z. Huijing and R. Shibasaki, "A vehicle-borne urban 3-D acquisition system using single-row laser range scanners," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on,* vol. 33, pp. 658-666, 2003.

[113] G. XueXian and Z. Sidong, "A Mobile System Using Lidar and Photogrammetry for Urban Spatial Objects Extraction," in *Information Engineering and Computer Science, 2009. ICIECS 2009. International Conference on*, 2009, pp. 1-4.

[114] E. Michaelsen and J. Meidow, "Stochastic reasoning for structural pattern recognition: An example from image-based UAV navigation," *Pattern Recognition,* vol. 47, pp. 2732-2744, 2014.

[115] J. Park, S. Im, K. H. Lee, and J. O. Lee, "Vision-Based SLAM System for Small UAVs in GPS-Denied Environments," *Journal of Aerospace Engineering,* vol. 25, pp. 519-529, Oct 2012.

[116] M. Smith, I. Posner, and P. Newman, "Adaptive compression for 3D laser data," *International Journal of Robotics Research,* vol. 30, pp. 914-935, Jun 2011.

[117] J. Artieda, J. M. Sebastian, P. Campoy, J. F. Correa, I. F. Mondragon, C. Martinez*, et al.*, "Visual 3-D SLAM from UAVs," *Journal of Intelligent & Robotic Systems,* vol. 55, pp. 299-321, Aug 2009.

[118] R. C. Gonzalez and R. E. Woods, "Digital image processing 3rd edition," ed: Prentice Hall, 2007.

[119] J. Canny, "A Computational Approach to Edge Detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. PAMI-8, pp. 679-698, 1986.

[120] M. 2016a, ed. Natick, MA: The Mathworks Inc., 2016.

[121] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica,* vol. 11, pp. 23-27, 1975.

[122] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *Systems, Man and Cybernetics, IEEE Transactions on,* vol. 9, pp. 62-66, 1979.

[123] M. Tuceryan, "Moment-based texture segmentation," *Pattern Recognition Letters,* vol. 15, pp. 659-668, 1994/07/01 1994.

[124] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern recognition,* vol. 24, pp. 1167-1186, 1991.

[125] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing textures in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3606-3613.

[126] A. Ellenberg, A. Kontsos, F. Moon, and I. Bartoli, "Bridge deck delamination identification from unmanned aerial vehicle infrared imagery," *Automation in Construction,* 2016.

[127] J. A. Feldman and Y. Yakimovsky, "Decision theory and artificial intelligence: I. A semantics-based region analyzer," *Artificial Intelligence,* vol. 5, pp. 349-371, 1974/12/01 1974.

[128] D. Ballard and C. Brown, "Computer Vision," ed: Prentice-Hall Inc., Englewood Cliffs, 1982.

[129] D. L. Pham, C. Xu, and J. L. Prince, "Current methods in medical image segmentation 1," *Annual review of biomedical engineering,* vol. 2, pp. 315-337, 2000.

[130] T. Yamaguchi and S. Hashimoto, "Fast crack detection method for large-size concrete surface images using percolation-based image processing," *Mach. Vision Appl.,* vol. 21, pp. 797-809, 2010.

[131] L. Álvarez, L. Baumela, P. Henríquez, and P. Márquez-Neila, "Morphological snakes," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, pp. 2197-2202.

[132] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision,* vol. 1, pp. 321-331, 1988.

[133] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern recognition,* vol. 13, pp. 111-122, 1981.

[134] R. Michel, R. Steinmeyer, M. Falk, and G. S. Harms, "A new detection algorithm for image analysis of single, fluorescence-labeled proteins in living cells," *Microscopy research and technique,* vol. 70, pp. 763-770, 2007.

[135] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR),* vol. 31, pp. 264-323, 1999.

[136] C. Bishop, "Pattern Recognition and Machine Learning (Information Science and Statistics), 1st edn. 2006. corr. 2nd printing edn," *Springer, New York,* 2007.

[137] R. Szeliski, *Computer vision: algorithms and applications*: Springer Science & Business Media, 2010.

[138] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision,* vol. 60, pp. 91-110, 2004.

[139] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer vision and image understanding,* vol. 110, pp. 346-359, 2008.

[140] E. Rosten and T. Drummond, "Machine Learning for High-Speed Corner Detection," in *Computer Vision – ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I*, A.

Leonardis, H. Bischof, and A. Pinz, Eds., ed Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 430-443.

[141] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, 2011, pp. 2564-2571.

[142] D. Mukherjee, Q. M. Jonathan Wu, and G. Wang, "A comparative experimental study of image feature detectors and descriptors," *Machine Vision and Applications,* vol. 26, pp. 443-466, 2015.

[143] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*: Cambridge university press, 2003.

[144] P. Harrington, *Machine learning in action* vol. 5: Manning Greenwich, CT, 2012.

[145] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *ACM SIGKDD explorations newsletter,* vol. 11, pp. 10-18, 2009.

[146] D. Lattanzi and G. Miller, "Robust Automated Concrete Damage Detection Algorithms for Field Applications," *Journal of Computing in Civil Engineering,* vol. 28, pp. 253-262, 2014.

[147] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," ed, 2007.

[148] L. Shapiro and G. C. Stockman, "Computer vision. 2001," *ed: Prentice Hall,* 2001.

[149] R. Szeliski, "Image alignment and stitching: A tutorial," *Foundations and Trends® in Computer Graphics and Vision,* vol. 2, pp. 1-104, 2006.

[150] A. Zhang, Q. Li, K. C. Wang, and S. Qiu, "Matched Filtering Algorithm for Pavement Cracking Detection," *Transportation Research Record: Journal of the Transportation Research Board,* vol. 2367, pp. 30-42, 2013.

[151] F. F. Feliciano, F. R. Leta, and F. B. Mainier, "Texture digital analysis for corrosion monitoring," *Corrosion Science,* vol. 93, pp. 138-147, 2015.

[152] C. Eschmann, C.-M. Kuo, C.-H. Kuo, and C. Boller, "High-resolution multisensor infrastructure inspection with unmanned aircraft systems," *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* vol. 1, pp. 125-129, 2013.

[153] T. Svoboda, J. Kybic, and V. Hlavac, *Image Processing, Analysis & and Machine Vision-A MATLAB Companion*: Thomson Learning, 2007.

[154] B. E. Bayer, "Color imaging array," ed: Google Patents, 1976.

[155] S. McHugh. (2016). *Camera Exposure*. Available: http://www.cambridgeincolour.com/

[156] Y.-F. Liu, S. Cho, B. Spencer Jr, and J.-S. Fan, "Concrete Crack Assessment Using Digital Image Processing and 3D Scene Reconstruction," *Journal of Computing in Civil Engineering*, 2014.

[157] J. W. Bagley, *The use of the panoramic camera in topographic surveying: with notes on the application of photogrammetry to aerial surveys* vol. 657: US Government Printing Office, 1917.

[158] J. Y. Bouguet. (2008). *Camera Calibration Toolbox for MATLAB*.

[159] J.-Y. Bouguet. (2013). *Camera calibration toolbox for matlab*. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/index.html#parameters

[160] J. Heikkila and O. Silvén, "A four-step camera calibration procedure with implicit image correction," in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, 1997, pp. 1106-1112.

[161] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence,* vol. 22, pp. 1330-1334, 2000.

[162] J. Valença, D. Dias-da-Costa, E. Júlio, H. Araújo, and H. Costa, "Automatic crack monitoring using photogrammetry and image processing," *Measurement,* vol. 46, pp. 433-441, 1// 2013.

[163] R. Jiang and D. V. Jauregui, "Development of a digital close-range photogrammetric bridge deflection measurement system," *Measurement,* vol. 43, pp. 1431-1438, 12// 2010.

[164] D. V. Jáuregui, K. R. White, C. B. Woodward, and K. R. Leitch, "Noncontact photogrammetric measurement of vertical bridge deflection," *Journal of Bridge Engineering,* vol. 8, pp. 212-222, 2003.

[165] L. Barazzetti and M. Scaioni, "Development and implementation of image-based algorithms for measurement of deformations in material testing," *Sensors,* vol. 10, pp. 7469-7495, 2010.

[166] J. Baqersad, P. Poozesh, C. Niezrecki, and P. Avitabile, "Photogrammetry and optical methods in structural dynamics – A review," *Mechanical Systems and Signal Processing*.

[167] R. I. Hartley and P. Sturm, "Triangulation," *Computer vision and image understanding,* vol. 68, pp. 146-157, 1997.

[168] E. B. Tadmor, R. E. Miller, and R. S. Elliott, *Continuum mechanics and thermodynamics: from fundamental concepts to governing equations*: Cambridge University Press, 2012.

[169] T. Luhmann, S. Robson, S. Kyle, and I. Harley, *Close range photogrammetry: Principles, methods and applications*: Whittles, 2006.

[170] J. Baqersad, P. Poozesh, C. Niezrecki, and P. Avitabile, "Photogrammetry and optical methods in structural dynamics – A review," *Mechanical Systems and Signal Processing,* vol. 86, Part B, pp. 17-34, 3/1/ 2017.

[171] D. Kalpoe, K. Khoshelham, and B. Gorte, "Vibration measurement of a model wind turbine using high speed photogrammetry," in *SPIE Optical Metrology*, 2011, pp. 80850J-80850J-11.

[172] Photometrix, "iWitnessPRO V3," ed. Australia: Photometrix Photogrammetry Software, 2017.

[173] GOM, "TRITOP Software User Manual," 6.2 ed. Widen, Switzerland: GOM Int., 2009.

[174] A. Ellenberg, L. Branco, A. Krick, I. Bartoli, and A. Kontsos, "Use of Unmanned Aerial Vehicle for Quantitative Infrastructure Evaluation," *Journal of Infrastructure Systems,* p. 04014054, 2014.

[175] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM,* vol. 24, pp. 381-395, 1981.

[176] C. Wu, "VisualSFM: A visual structure from motion system," 2011.

[177] G. Zhang, P. Vela, and I. Brilakis, "Automatic generation of as-built geometric civil infrastructure models from point cloud data," *Computing in Civil and Building Engineering,* pp. 406-413, 2014.

[178] B. Horn, *Robot vision*: MIT press, 2010.

[179] B. Pan, K. M. Qian, H. M. Xie, and A. Asundi, "Two-dimensional digital image correlation for in-plane displacement and strain measurement: a review," *Measurement Science & Technology,* vol. 20, Jun 2009.

[180] M. A. Sutton, J. J. Orteu, and H. Schreier, *Image correlation for shape, motion and deformation measurements: basic concepts, theory and applications*: Springer Science & Business Media, 2009.

[181] B. Wisner and A. Kontsos, "Fatigue Damage Precursor Identification Using Nondestructive Evaluation Coupled with Electron Microscopy," in *Fracture, Fatigue, Failure and Damage Evolution, Volume 8: Proceedings of the 2016*

Annual Conference on Experimental and Applied Mechanics, A. T. Zehnder, J. Carroll, K. Hazeli, R. B. Berke, G. Pataky, M. Cavalli, *et al.*, Eds., ed Cham: Springer International Publishing, 2017, pp. 1-8.

[182] S. Rajaram, P. A. Vanniamparambil, F. Khan, M. Bolhassani, A. Koutras, I. Bartoli, *et al.*, "Full-field deformation measurements during seismic loading of masonry buildings," *Structural Control and Health Monitoring,* vol. 24, pp. e1903-n/a, 2017.

[183] P. A. Vanniamparambil, I. Bartoli, K. Hazeli, J. Cuadra, E. Schwartz, R. Saralaya, *et al.*, "An integrated structural health monitoring approach for crack growth monitoring," *Journal of Intelligent Material Systems and Structures,* vol. 23, pp. 1563-1573, 2012.

[184] B. Pan, Z. Lu, and H. Xie, "Mean intensity gradient: An effective global parameter for quality assessment of the speckle patterns used in digital image correlation," *Optics and Lasers in Engineering,* vol. 48, pp. 469-477, 4// 2010.

[185] D. Lecompte, H. Sol, J. Vantomme, and A. Habraken, "Analysis of speckle patterns for deformation measurements by DIC," *Proceedings of SPIE Vol. 6341,* pp. E1-E6, 2006.

[186] S. Yaofeng and J. H. Pang, "Study of optimal subset size in digital image correlation of speckle pattern images," *Optics and lasers in engineering,* vol. 45, pp. 967-974, 2007.

[187] G. Crammond, S. W. Boyd, and J. M. Dulieu-Barton, "Speckle pattern quality assessment for digital image correlation," *Optics and Lasers in Engineering,* vol. 51, pp. 1368-1378, 12// 2013.

[188] S. Park, H. Park, J. Kim, and H. Adeli, "3D displacement measurement model for health monitoring of structures using a motion capture system," *Measurement,* vol. 59, pp. 352-362, 2015.

[189] A. Sabato and C. Niezrecki, "Feasibility of digital image correlation for railroad tie inspection and ballast support assessment," *Measurement,* vol. 103, pp. 93-105, 6// 2017.

[190] B. Jafari, A. Khaloo, and D. Lattanzi, "Long-term monitoring of structures through point cloud analysis," in *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, 2016, pp. 98052K-98052K-8.

[191] G. A. Shaw and H.-h. K. Burke, "Spectral imaging for remote sensing," *Lincoln Laboratory Journal,* vol. 14, pp. 3-28, 2003.

[192] R. Bridgelall, J. B. Rafert, and D. Tolliver, "Hyperspectral imaging utility for transportation systems," in *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, 2015, pp. 943522-943522-14.

[193] G. Butcher and J. Mottar, *Tour of the electromagnetic spectrum*: National Aeronautics and Space Administration, 2010.

[194] D. Malacara-Hernández and Z. Malacara-Hernández, *Handbook of optical design*: CRC Press, 2013.

[195] K. Sayanagi, "Pinhole Imagery*," *Journal of the Optical Society of America,* vol. 57, pp. 1091-1098, 1967/09/01 1967.

[196] B. Pan, D. Wu, Z. Wang, and Y. Xia, "High-temperature digital image correlation method for full-field deformation measurement at 1200 C," *Measurement science and technology,* vol. 22, p. 015701, 2010.

[197] K. Mangold, J. A. Shaw, and M. Vollmer, "The physics of near-infrared photography," *European Journal of physics,* vol. 34, p. S51, 2013.

[198] K. K. Mustakallio and P. Korhonen, "Monochromatic Ultraviolet-Photography in Dermatology** From the Department of Dermatology, University Central Hospital, Snellmaninkatu 14, Helsinki 17, Finland and from the Department of Photography, University of Helsinki, Helsinki, Finland," *Journal of Investigative Dermatology,* vol. 47, pp. 351-356, 1966.

[199] R. Usamentiaga, P. Venegas, J. Guerediaga, L. Vega, J. Molleda, and F. G. Bulnes, "Infrared thermography for temperature measurement and non-destructive testing," *Sensors,* vol. 14, pp. 12305-12348, 2014.

[200] S. Bagavathiappan, B. Lahiri, T. Saravanan, J. Philip, and T. Jayakumar, "Infrared thermography for condition monitoring–a review," *Infrared Physics & Technology,* vol. 60, pp. 35-55, 2013.

[201] A. Kylili, P. A. Fokaides, P. Christou, and S. A. Kalogirou, "Infrared thermography (IRT) applications for building diagnostics: A review," *Applied Energy,* vol. 134, pp. 531-549, 2014.

[202] T. L. Bergman, F. P. Incropera, D. P. DeWitt, and A. S. Lavine, *Fundamentals of Heat and Mass Transfer*: Wiley, 2011.

[203] F. Khan, M. Bolhassani, A. Kontsos, A. Hamid, and I. Bartoli, "Modeling and experimental implementation of infrared thermography on concrete masonry structures," *Infrared Physics & Technology,* vol. 69, pp. 228-237, 2015.

[204] ASTM, "Standard Test Method for Detecting Delaminations in Bridge Decks Using Infrared Thermography," ed. West Conshohocken, PA: ASTM International, 2007.

[205] M. R. Clark, D. M. McCann, and M. C. Forde, "Application of infrared thermography to the non-destructive testing of concrete and masonry bridges," *NDT & E International,* vol. 36, pp. 265-275, 6// 2003.

[206] K. Kurita, M. Oyado, H. Tanaka, and S. Tottori, "Active infrared thermographic inspection technique for elevated concrete structures using remote heating system," *Infrared Physics & Technology,* vol. 52, pp. 208-213, 2009.

[207] C. Liu, Y. Liu, H. Wu, and R. Dong, "A Safe Flight Approach of the UAV in the Electrical Line Inspection," *International Journal of Emerging Electric Power Systems,* vol. 16, pp. 503-515, 2015.

[208] J. Cassella and J. Cavalier, "Method of detecting and repairing a structural roof damaged by subsurface moisture," ed: Google Patents, 1974.

[209] P. A. Fokaides and S. A. Kalogirou, "Application of infrared thermography for the determination of the overall heat transfer coefficient (U-Value) in building envelopes," *Applied Energy,* vol. 88, pp. 4358-4365, 2011.

[210] Z. Li, W. Yao, S. Lee, C. Lee, and Z. Yang, "Application of infrared thermography technique in building finish evaluation," *Journal of nondestructive evaluation,* vol. 19, pp. 11-19, 2000.

[211] M. Scaioni, T. Feng, L. Barazzetti, M. Previtali, and R. Roncella, "Image-based deformation measurement," *Applied Geomatics,* vol. 7, pp. 75-90, 2014.

[212] C. Koch, Z. Zhu, S. G. Paal, and I. Brilakis, "Machine Vision Techniques for Condition Assessment of Civil Infrastructure," in *Integrated Imaging and Vision Techniques for Industrial Inspection*, ed: Springer, 2015, pp. 351-375.

[213] Z. Zhu, S. German, and I. Brilakis, "Visual retrieval of concrete crack properties for automated post-earthquake structural safety evaluation," *Automation in Construction,* vol. 20, pp. 874-883, 11// 2011.

[214] M. R. Jahanshahi and S. F. Masri, "Adaptive vision-based crack detection using 3D scene reconstruction for condition assessment of structures," *Automation in Construction,* vol. 22, pp. 567-576, 2012.

[215] K. Cunningham, D. Lattanzi, R. Dell'Andrea, M. Riley, T. Huette, R. Goetz*, et al.*, "UAS-Based Inspection of the Placer River Trail Bridge: A Data-Driven Approach," in *Structures Congress 2015*, 2015, pp. 607-615.

[216] D. Lattanzi and G. Miller, "Review of Robotic Infrastructure Inspection Systems," *Journal of Infrastructure Systems,* p. 04017004, 2017.

[217] (2016). *Structural inspection*. Available: http://airfilmsinspections.com/industrial-inspections/structural-inspection/

[218] D. Mascarenas, E. Flynn, C. Farrar, G. Park, and M. Todd, "A Mobile Host Approach for Wireless Powering and Interrogation of Structural Health Monitoring Sensor Networks," *Ieee Sensors Journal,* vol. 9, pp. 1719-1726, Dec 2009.

[219] D. Mascareñas, E. Flynn, M. Todd, G. Park, and C. Farrar, "Wireless sensor technologies for monitoring civil structures," *Sound and Vibration,* vol. 42, pp. 16-21, 2008.

[220] (2015). *Microdrones*. Available: https://www.microdrones.com/en/home/

[221] J. Torres-Sánchez, F. López-Granados, and J. Peña, "An automatic object-based method for optimal thresholding in UAV images: Application for vegetation detection in herbaceous crops," *Computers and Electronics in Agriculture,* vol. 114, pp. 43-52, 2015.

[222] C. Sikorsky, "Identification of gaps in structural health monitoring technologies for bridges," ed: Citeseer, 1999.

[223] B. Glisic. (11/4). *SHM DATA MANAGEMENT*. Available: http://www.princeton.edu/~bglisic/Research.html

[224] F. Moon, J. Devitis, D. Masceri, A. E. Aktan, B. Buchter, B. Basily*, et al.*, "Self-contained rapid modal testing system for highway bridges," 2015.

[225] Microsoft. (2015). *Microsoft Image Composite Editor*. Available: http://research.microsoft.com/en-us/um/redmond/projects/ice/

[226] C. Team, "CHDK User Manual," ed, 2012.

[227] C. Petiot, S. Roux, and H. Francois, "Multi-scale method for measuring the shape, movement and/or deformation of a structural part subjected to stresses by creating a plurality of colorimetric speckled patterns," ed: Google Patents, 2014.

[228] (2014). *BP and AeroVironment Launch First FAA-Approved, Commercial Unmanned Aircraft Operations Over Land and Water; Providing Comprehensive GIS Services*. Available: http://www.avinc.com/resources/press_release/bp-and-aerovironment-launch-first-faa-approved-commercial-unmanned-aircraft

[229] A. Ellenberg, A. Kontsos, F. Moon, and I. Bartoli, "Rapid, Preliminary Bridge Deck Damage Identification from Unmanned Aerial System Imagery," presented at the NDE/NDT for Highways and Bridges: Structural Materials Technology, Portland, OR, 2016.

[230] H. Zakeri, F. M. Nejad, and A. Fahimifar, "Image Based Techniques for Crack Detection, Classification and Quantification in Asphalt Pavement: A Review," *Archives of Computational Methods in Engineering,* pp. 1-43, 2016// 2016.

[231] P. M. Dare, H. B. Hanley, C. S. Fraser, B. Riedel, and W. Niemeier, "An operational application of automatic feature extraction: The measurement of cracks in concrete structures," *Photogrammetric Record,* vol. 17, pp. 453-464, Apr 2002.

[232] S. Chen, B. Lin, X. Han, and X. Liang, "Automated inspection of engineering ceramic grinding surface damage based on image recognition," *The International Journal of Advanced Manufacturing Technology,* vol. 66, pp. 431-443, 2013/04/01 2013.

[233] Z. Yongxia, W. Guoqiang, and Z. Chuncheng, "A Novel Image Segmentation Method of Pavement Surface Cracks Based on Fractal Theory," in *Computational Intelligence and Security, 2008. CIS '08. International Conference on*, 2008, pp. 485-488.

[234] A. Ellenberg, A. Kontsos, I. Bartoli, and A. Pradhan, "Masonry Crack Detection Application of an Unmanned Aerial Vehicle," *Computing in Civil and Building Engineering,* 2014.

[235] H. Dongna, T. Tian, Y. Hengxiang, X. Shibo, and W. Xiujin, "Wall crack detection based on image processing," in *Intelligent Control and Information Processing (ICICIP), 2012 Third International Conference on Intelligent Control and Information Processing*, 2012, pp. 597-600.

[236] L. Pei, Z. Xie, and J. Dai, "Joint edge detector based on Laplacian pyramid," in *Image and Signal Processing (CISP), 2010 3rd International Congress on*, 2010, pp. 978-982.

[237] S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, and M. Goldbaum, "Detection of blood vessels in retinal images using two-dimensional matched filters," *IEEE Transactions on medical imaging,* vol. 8, pp. 263-269, 1989.

[238] A. Zhang, Q. Li, K. C. P. Wang, and S. Qiu, "Matched Filtering Algorithm for Pavement Cracking Detection," *Transportation Research Record,* pp. 30-42, 2013.

[239] Q. Zou, Y. Cao, Q. Li, Q. Mao, and S. Wang, "CrackTree: Automatic crack detection from pavement images," *Pattern Recognition Letters,* vol. 33, pp. 227-238, 2/1/ 2012.

[240] J. Huang, W. Liu, and X. Sun, "A Pavement Crack Detection Method Combining 2D with 3D Information Based on Dempster-Shafer Theory," *Computer-Aided Civil and Infrastructure Engineering,* vol. 29, pp. 299-313, 2014.

[241] M. O'Byrne, F. Schoefs, B. Ghosh, and V. Pakrashi, "Texture Analysis Based Damage Detection of Ageing Infrastructural Elements," *Computer-Aided Civil and Infrastructure Engineering,* vol. 28, pp. 162-177, Mar 2013.

[242] H. Son, N. Hwang, C. Kim, and C. Kim, "Rapid and automated determination of rusted surface areas of a steel bridge for robotic maintenance systems," *Automation in Construction,* vol. 42, pp. 13-24, 6// 2014.

[243] K. Máthé and L. Buşoniu, "Vision and control for UAVs: A survey of general methods and of inexpensive platforms for infrastructure inspection," *Sensors,* vol. 15, pp. 14887-14916, 2015.

[244] L. Otero, "Remote Sensing with Mobile LiDAR and Imaging Sensors for Railroad Bridge Inspections," *Transportation Research Board New Ideas for Rail Safety,* 2016.

[245] C. M. Yeum and S. J. Dyke, "Vision-Based Automated Crack Detection for Bridge Inspection," *Computer-Aided Civil and Infrastructure Engineering,* 2015.

[246] D. Reagan, A. Sabato, C. Niezrecki, T. Yu, and R. Wilson, "An autonomous unmanned aerial vehicle sensing system for structural health monitoring of bridges," in *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, 2016, pp. 980414-980414-9.

[247] D. Reagan, A. Sabato, and C. Niezrecki, "Unmanned aerial vehicle acquisition of three-dimensional digital image correlation measurements for structural health monitoring of bridges," in *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, 2017, pp. 1016909-1016909-10.

[248] R. J. Dobson, C. Brooks, C. Roussi, and T. Colling, "Developing an unpaved road assessment system for practical deployment with high-resolution optical data collection using a helicopter UAV," in *Unmanned Aircraft Systems (ICUAS), 2013 International Conference on*, 2013, pp. 235-243.

[249] C. Roussi, C. Brooks, T. Colling, R. Dobson, M. K. Watkins, D. Dean*, et al.*, "Characterization of Unpaved Road Condition Through the Use of Remote Sensing Project," Michigan Technological University, Houghton, MI 2014.

[250] R. McLassus, "Rust and dirt," ed: Wikimedia Commons, 2006.

[251] A. Ellenberg, A. Kontsos, F. Moon, and I. Bartoli, "Bridge related damage quantification using unmanned aerial vehicle imagery," *Structural Control and Health Monitoring,* vol. 23, pp. 1168-1179, Sep 2016.

[252] P. A. Vanniamparambil, M. Bolhassani, R. Carmi, F. Khan, I. Bartoli, F. L. Moon*, et al.*, "A data fusion approach for progressive damage quantification in reinforced concrete masonry walls," *Smart Materials and Structures,* vol. 23, p. 015007, 2014.

[253] V. Blåsjö, "The isoperimetric problem," *The American Mathematical Monthly,* vol. 112, pp. 526-566, 2005.

[254] http://www.boggsinspect.com/system/files/images/227%20Maple%20Park%20Ave%20006.jpg.

[255] http://everdryatlanta.com/wp-content/uploads/2014/12/atlanta-foundation-624x468.jpg.

[256] http://rcpblock.com/articles/wp-content/uploads/2013/04/cracked-concrete.jpg.

[257] http://wallsnceiling.com/wp-content/uploads/2013/09/9-Cracked_Wall_well-fix-it-for-you.jpg.

[258] fc00.deviantart.net/fs11/i/2006/228/8/0/Brick_wall_by_ashzstock.jpg.

[259] fc06.deviantart.net/fs46/f/2009/170/4/9/Concrete_Wall_86_by_dknucklesstock.jpg.

[260] http://dev.nimrodtsabari.net/assets/concrete-1-1350.jpg.

[261] www.rgbstock.com/bigphoto/oot6lGg/cement+textures+1.

[262] N. Gucunski, H. Ghasemi, and A. Maher, "Condition Monitoring of Concrete Bridge Decks Through Periodical NDE Using Multiple Technologies," in *Smart Monitoring, Assessment and Rehabilitation of Civil Structures - SMAR 2011*, Dubai, UAE, 2011, 2011.

[263] W. Tobiasson and C. Korhonen, "Summary of Corps of Engineers Research on Roof Mositure Detection and the Thermal Resistance of Wet Insulation," DTIC Document1978.

[264] D. Titman, "Applications of thermography in non-destructive testing of structures," *NDT & E International,* vol. 34, pp. 149-154, 2001.

[265] R. Montero, J. G. Victores, S. Martínez, A. Jardón, and C. Balaguer, "Past, present and future of robotic tunnel inspection," *Automation in Construction,* vol. 59, pp. 99-112, 11// 2015.

[266] C. Ibarra-Castanedo, M. Genest, S. Guibert, J.-M. Piau, X. P. Maldague, and A. Bendada, "Inspection of aerospace materials by pulsed thermography, lock-in thermography, and vibrothermography: a comparative study," in *Defense and Security Symposium*, 2007, pp. 654116-654116-9.

[267] G. F. Pla-Rucki and M. O. Eberhard, "Imaging of reinforced concrete: State-of-the-art review," *Journal of infrastructure systems,* vol. 1, pp. 134-141, 1995.

[268] S. Mills, J. Ford, and L. Mejías, "Vision Based Control for Fixed Wing UAVs Inspecting Locally Linear Infrastructure Using Skid-to-Turn Maneuvers," *Journal of Intelligent & Robotic Systems,* vol. 61, pp. 29-42, 2011/01/01 2011.

[269] S. Harwin and A. Lucieer, "Assessing the Accuracy of Georeferenced Point Clouds Produced via Multi-View Stereopsis from Unmanned Aerial Vehicle (UAV) Imagery," *Remote Sensing,* vol. 4, pp. 1573-1599, Jun 2012.

[270] H. Eisenbeiss and M. Sauerbier, "Investigation of uav systems and flight modes for photogrammetric applications," *Photogrammetric Record,* vol. 26, pp. 400-421, Dec 2011.

[271] M. Scott and D. Kruger, "Infrared thermography as a diagnostic tool for subsurface assessments of concrete structures," 2014.

[272] M. Matsumoto, K. Mitani, and F. N. Catbas, "NDE for Bridge Assessment using Image Processing and Infrared Thermography 2."

[273] K. Vaghefi, D. K. Harris, and C. N. Brooks, "Combined imaging technologies for concrete bridge deck condition assessment," *Journal of Performance of Constructed Facilities,* 2013.

[274] L. Merino, F. Caballero, J. R. Martínez-de Dios, J. Ferruz, and A. Ollero, "A cooperative perception system for multiple UAVs: Application to automatic detection of forest fires," *Journal of Field Robotics,* vol. 23, pp. 165-184, 2006.

[275] S.-H. Kee and N. Gucunski, "Interpretation of Flexural Vibration Modes from Impact-Echo Testing," *Journal of Infrastructure Systems,* p. 04016009, 2016.

[276] S. Yahyanejad, J. Misiorny, and B. Rinner, "Lens distortion correction for thermal cameras to improve aerial imaging with small-scale UAVs," in *Robotic and Sensors Environments (ROSE), 2011 IEEE International Symposium on*, 2011, pp. 231-236.

[277] M. 2014b, ed. Natick, MA: The Mathworks Inc., 2014.

[278] S.-H. Kee, T. Oh, J. S. Popovics, R. W. Arndt, and J. Zhu, "Nondestructive bridge deck testing with air-coupled impact-echo and infrared thermography," *Journal of Bridge Engineering,* vol. 17, pp. 928-939, 2011.

[279] F. Khan, A. Ellenberg, S. Ye, A. E. Aktan, F. Moon, A. Kontsos*, et al.*, "Multispectral aerial imaging for infrastructure evaluation," in *NDE/NDT for Structural Materials Technology for Highway & Bridges*, 2014, pp. 123-130.

[280] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden, "Pyramid methods in image processing," *RCA engineer,* vol. 29, pp. 33-41, 1984.

[281] I. Clarke, G. Sellers, and Z. Yusuf, "System and method for creating a panorama image from a plurality of source images," ed: Google Patents, 2003.

[282] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *International journal of computer vision,* vol. 74, pp. 59-73, 2007.

[283] X. Zhao, H. Liu, Y. Yu, X. Xu, W. Hu, M. Li, *et al.*, "Bridge Displacement Monitoring Method Based on Laser Projection-Sensing Technology," *Sensors (Basel, Switzerland),* vol. 15, pp. 8444-8463, 2015.

[284] S.-i. Nakamura, "GPS measurement of wind-induced suspension bridge girder displacements," *Journal of Structural Engineering,* vol. 126, pp. 1413-1419, 2000.

[285] P.-J. Bristeau, F. Callou, D. Vissière, and N. Petit, "The navigation and control technology inside the ar. drone micro uav," in *18th IFAC world congress*, 2011, pp. 1477-1484.

[286] http://ardrone2.parrot.com/. (2013). *AR Drone 2.0*.

[287] GOM, "TRITOP v6.2 User Manual-Software," ed, 2009.

[288] GOMmbH, "ARAMIS," 6.3 ed. Braunschweig, Germany: GOM mbH, 2009.

[289] K. Khoshelham and S. O. Elberink, "Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications," *Sensors,* vol. 12, pp. 1437-1454, 2012.

[290] V. LiDAR, "VPL-16," ed, 2016.

[291] T. Sieberth, R. Wackrow, and J. H. Chandler, "UAV Image Blur-Its Influence and Ways to Correct it," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences,* vol. 40, p. 33, 2015.

[292] R. Chase. (2014). *Dirt Pile Blamed for Damage to I-495 Bridge*.

[293] A. R. Greenleaf, *Photographic optics*: Macmillan, 1950.

[294] P. S. Yukihiro Ito. (2015). *SMARTSENSYS*.

[295] B. Moeslund Thomas, "Introduction to video and image processing," ed: London: Springer-Verlag, 2012.

[296] N. E. Dowling, *Mechanical Behavior of Materials: Engineering Methods for Deformation, Fracture, and Fatigue*: Pearson, 2013.

[297] J. Blaber, "Ncorr digital image correlation software," ed: Georgia Institute of Technology, 2014.

[298] J. Blaber, B. Adair, and A. Antoniou, "Ncorr: open-source 2D digital image correlation matlab software," *Experimental Mechanics,* vol. 55, pp. 1105-1122, 2015.

[299] M. N. Gillins, D. T. Gillins, and C. Parrish, "Cost-Effective Bridge Safety Inspections Using Unmanned Aircraft Systems (UAS)," in *Geotechnical and Structural Engineering Congress 2016*, pp. 1931-1940.

[300] N. Tyutyundzhiev, K. Lovchinov, F. Martínez-Moreno, J. Leloux, and L. Narvarte, "Advanced PV modules inspection using multirotor UAV," 2015.

[301] B. Cohen, S. Ye, G. Karaman, F. Khan, I. Bartoli, A. Pradhan*, et al.*, "Design and Implementation of an Integrated Operations and Preservation Performance Monitoring System for Asset Management of Major Bridges," in *EWSHM-7th European Workshop on Structural Health Monitoring*, 2014.

[302] X. Z. Fan, E. Pomerantseva, M. Gnerlich, A. Brown, K. Gerasopoulos, M. McCarthy*, et al.*, "Tobacco mosaic virus: A biological building block for micro/nano/bio systems," *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films,* vol. 31, p. 050815, 2013.

[303] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The grasp multiple micro-uav testbed," *IEEE Robotics & Automation Magazine,* vol. 17, pp. 56-65, 2010.

[304] A. Belkadi, L. Ciarletta, and D. Theilliol, "UAVs fleet control design using distributed particle swarm optimization: A leaderless approach," in *Unmanned Aircraft Systems (ICUAS), 2016 International Conference on*, 2016, pp. 364-371.

[305] D. Mascarenas, Y. Yang, A. Green, E. Flynn, and C. Farrar, "Robotic Research at the LANL Engineering Institute," Los Alamos National Laboratory (LANL)2016.

[306] N. Gucunski, S.-H. Kee, H. La, B. Basily, and A. Maher, "Delamination and concrete quality assessment of concrete bridge decks using a fully autonomous RABIT platform," *Structural Monitoring and Maintenance,* vol. 2, pp. 19-34, 2015.

[307] P. Prasanna, K. J. Dana, N. Gucunski, B. B. Basily, H. M. La, R. S. Lim*, et al.*, "Automated Crack Detection on Concrete Bridges," *IEEE Transactions on Automation Science and Engineering,* vol. 13, pp. 591-599, 2016.

[308] ABAQUS, 6.13 ed: Dassault Systemes, 2013.

[309] C. M. Barr, P. M. Gallagher, B. M. Wadzuk, and A. L. Welker, "Water Quality Impacts of Green Roofs Compared with Other Vegetated Sites," *Journal of Sustainable Water in the Built Environment,* p. 04017007, 2017.

[310] P. Tokarczyk, J. P. Leitao, J. Rieckermann, K. Schindler, and F. Blumensaat, "High-quality observation of surface imperviousness for urban runoff modelling using UAV imagery," *Hydrology and Earth System Sciences,* vol. 19, pp. 4215-4228, 2015.

[311] P. A. Vanniamparambil, "A novel cross-validated nondestructive evaluation framework for damage detection using acoustic emission," Drexel University, 2014.