

# **Contactless Displacement Measurement for Vision-based Structural Health Monitoring Using Optical Flow**

---

A Thesis

Presented to

the faculty of the School of Engineering and Applied Science

University of Virginia

---

in partial fulfillment  
of the requirements for the degree

Master of Science

by

Savannah J. Washlesky

May 2019

# APPROVAL SHEET

This Thesis  
is submitted in partial fulfillment of the requirements  
for the degree of  
Master of Science

Author Signature: \_\_\_\_\_

This Thesis has been read and approved by the examining committee:

Advisor: Dr. Devin K. Harris

Committee Member: Dr. Jose Gomez

Committee Member: Dr. Arsalan Heydarian

Committee Member: \_\_\_\_\_

Committee Member: \_\_\_\_\_

Committee Member: \_\_\_\_\_

Accepted for the School of Engineering and Applied Science:



Craig H. Benson, School of Engineering and Applied Science

May 2019

## **Abstract**

Inspection and assessment are necessary steps to maintain and ensure the safety and capacity of critical infrastructure systems such as bridges and other civil structures. Traditional sensors, including accelerometers, strain gauges, and string potentiometers, can be difficult to place in remote, offshore, and complex environments where the structure is very tall or hard to reach. Vision-based systems offer an innovative approach for structural health monitoring because they do not require contact with the target structure. This work proposes a contactless vision-based system for structural displacement measurement with real-time processing potential. Image acquisition and processing was implemented on an NVIDIA Jetson TX2 connected to a CMOS camera using the Lucas-Kanade-based optical flow method for feature tracking. A series of lab tests studied the performance of the system under a variety of targets, lighting conditions, and motion frequencies to validate the tracking algorithm against a cross-correlation method and a traditional displacement measurement device. Field testing measured the end displacement of a traffic signal mast arm and results show that the system captured movement on a subpixel level. With confirmation of feasibility of the proposed system to provide contactless vision-based displacement measurement through feature tracking, recommended steps for deployment of the system in the built environment for short-term monitoring of bridges or traffic signal structures will be offered.

Keywords: structural health monitoring, displacement measurement, KLT-based optical flow

## **Acknowledgements**

This work would not have been possible without the support of many individuals, who include but are certainly not limited to the following. I would like to thank:

- Dr. Devin Harris, my advisor, for his constant encouragement and confidence in me, as well as for giving me the freedom to pursue my interests while growing in my knowledge of civil engineering over the past two years. My time in Charlottesville has been memorable and I am still appreciative of the opportunity to join your research group at UVa.
- Dr. Jose Gomez, for serving on my committee, sharing his knowledge of structural engineering, and championing all of his students and their passions.
- Dr. Arsalan Heydarian, for serving on my committee and recruiting friendly students.
- The MOB Lab Research Group, for their collaboration, camaraderie, and support, especially those that assisted me during my testing, specifically Hong Liang, Ciara Horne, and Benjamin Griffith, as well as Mohamad Alipour for his mentorship.
- ESE department personnel, for their help with implementing my lab work and field tests and providing technical support, specifically Keegan Gumbs and David Thomas.
- My friends in the ESE department and the Link Lab, for making my time in the office more enjoyable and creating outlets to live a balanced life through the cherished happy hours, hikes, runs, and soccer matches.
- My professors and friends from Notre Dame, for building my foundation not only as a civil engineer, but as a global citizen, a leader, and a friend, specifically Dr. Tracy Kijewski-Correa, Dr. Ashley Thrall, Dr. Maria Gibbs, and my McGlenn sisters. Even though I'm a Wahoo now, I'll always bleed blue and gold.
- My family, for giving me life, love, and laughter. I would not be the person I am today without their abundant support in encouraging me to realize my dreams each day at a time. This thesis is dedicated to them.

Finally, I express my gratitude to God, to whom I owe everything.

# Table of Contents

<b>Abstract .....</b>	<b>i</b>
<b>Acknowledgements.....</b>	<b>ii</b>
<b>Table of Contents .....</b>	<b>iii</b>
<b>List of Tables .....</b>	<b>vi</b>
<b>List of Figures.....</b>	<b>vii</b>
<b>1 Introduction.....</b>	<b>1</b>
1.1 Background and Motivation .....	1
1.2 Research Objectives .....	1
1.3 Thesis Outline .....	2
<b>2 Literature Review of Structural Health Monitoring.....</b>	<b>3</b>
2.1 Traditional Methods for Monitoring.....	3
2.2 Introduction of Vision-based SHM Methods .....	5
2.2.1 Applications of Vision-based SHM .....	5
2.2.2 Current Vision-based Displacement Systems.....	6
2.3 Background on Image Processing Methods .....	6
2.3.1 Area-based Template Matching.....	6
2.3.2 Feature-based Template Matching.....	7
2.4 Background on Computing Methods in SHM.....	8
2.4.1 Real-time vs. Post-processing.....	8
2.4.2 Introduction of GPUs for Computing .....	9
<b>3 Experimental Methods.....</b>	<b>11</b>
3.1 Motivation .....	11
3.2 Image Acquisition.....	11
3.2.1 Computing Device .....	11

3.2.2	Camera and Lens.....	14
3.3	Image Processing for Displacement Measurement.....	20
3.3.1	Open-source and Commercial Software.....	20
3.3.2	Normalized Cross-Correlation.....	22
3.3.3	Optical Flow Method.....	24
<b>4</b>	<b>Experimental Validation.....</b>	<b>30</b>
4.1	Motivation.....	30
4.2	Final System Design.....	30
4.3	Lab Testing.....	31
4.3.1	Feasibility Study.....	31
4.3.2	Lighting Study.....	35
4.3.3	Natural Feature Tracking Study.....	36
4.3.4	Comparison Against Traditional Displacement Measurement.....	38
4.4	Field Testing.....	41
4.4.1	Traffic Signal Structure Deflection Study.....	41
<b>5</b>	<b>Experimental Results.....</b>	<b>45</b>
5.1	Lab Testing Results.....	45
5.1.1	Feasibility Study Results.....	45
5.1.2	Lighting Study Results.....	52
5.1.3	Natural Feature Tracking Study Results.....	57
5.1.4	Comparison Against Traditional Displacement Measurement Results.....	60
5.2	Field Testing Results.....	63
5.2.1	Traffic Signal Structure Deflection Study Results.....	63
<b>6</b>	<b>Conclusion.....</b>	<b>66</b>
6.1	Future Work.....	67

6.1.1	Physical Requirements .....	67
6.1.2	Computational Capabilities .....	71
<b>References</b>	.....	<b>72</b>
<b>Appendix A</b>	.....	<b>81</b>

## List of Tables

Table 3.1. Raspberry Pi 3 specifications. ....	12
Table 3.2. Comparison of specifications between NVIDIA Jetson TX1 and TX2.....	13
Table 3.3. Raspberry Pi camera module specifications.....	16
Table 3.4. NVIDIA Jetson TX2 camera module specifications. ....	17
Table 3.5. GoPro Hero 3+ camera specifications. ....	18
Table 3.6. Tamron A17 lens specifications. ....	18
Table 3.7. FLIR Grasshopper3 camera specifications. ....	19
Table 3.8. Tamron 672D lens specifications. ....	20
Table 3.9. Commercial software for image processing.....	20
Table 3.10. Open-source software for image processing. ....	21
Table 4.1. Zaber translation stage specifications. ....	33
Table 4.2. Feasibility test matrix.....	35
Table 4.3. Lighting test matrix.....	36
Table 4.4. Natural feature test matrix.....	38
Table 4.5. String pot comparison test matrix.....	41
Table 4.6. Traffic signal structure test camera acquisition periods.....	44
Table 5.1. Iteration of Lucas-Kanade-based optical flow method processing parameters.....	46
Table 5.2. Feasibility test: Average percent correlation between NCC and OFM displacement results.....	51
Table 5.3. Lighting test: Average percent correlation between NCC and OFM displacement results. ....	57

## List of Figures

Figure 2.1. Trackable structural features.....	8
Figure 2.2. CPU vs. GPU bandwidth trends from 2003 to present (adapted from NVIDIA Corporation 2018).....	9
Figure 3.1. Raspberry Pi 3 Model B.....	12
Figure 3.2. NVIDIA Jetson TX2.....	13
Figure 3.3. (a) Field of view estimation and (b) Image dimensions.....	15
Figure 3.4. Raspberry Pi camera module with pan-and-tilt mechanism.....	16
Figure 3.5. (a) GoPro Hero 3+ and (b) Tamron A17.....	18
Figure 3.6. (a) FLIR Grasshopper3 and (b) Tamron 672D.....	19
Figure 3.7. Digital image correlation process.....	23
Figure 3.8. Optical flow estimation of an object through space.....	25
Figure 3.9. Shi-Tomasi corner detection using threshold values (adapted from Intel Corporation 2019).....	26
Figure 3.10. Tracking of an object through space using optical flow.....	28
Figure 3.11. Feature tracking across a set of images using optical flow.....	28
Figure 4.1. System process diagram.....	31
Figure 4.2. Speckle pattern.....	32
Figure 4.3. Checkerboard pattern.....	33
Figure 4.4. Target cards mounted on translation stage.....	33
Figure 4.5. Feasibility test setup.....	34
Figure 4.6. Steel section (side and forward orientations) and concrete cylinder on translation stage.....	37
Figure 4.7. Extended string potentiometer supported by tripod under bridge girder.....	39
Figure 4.8. String pot comparison test setup: (a) Close-up of target and (b) Camera to target view.....	40
Figure 4.9. Plan view of Fontaine Ave at Ray C Hunt Dr (adapted from Google Maps 2019).....	42
Figure 4.10. Traffic signal structure test instrumentation setup.....	43
Figure 4.11. Traffic signal structure test setup with dimensions.....	43
Figure 5.1. Feasibility test: Sample optical flow tracking for speckle and checkerboard target patterns.....	47

Figure 5.2. Original ROI for speckle and checkerboard target patterns.....	49
Figure 5.3. Feasibility test: Sample horizontal displacement for speckle target pattern.....	49
Figure 5.4. Feasibility test: Sample horizontal displacement for checkerboard target pattern.....	50
Figure 5.5. Various aperture settings under ambient lighting.....	52
Figure 5.6. Various aperture settings under dark conditions.....	52
Figure 5.7. Lighting test: Sample optical flow tracking for speckle and checkerboard target patterns. ....	53
Figure 5.8. ROI iteration for speckle and checkerboard target patterns.....	54
Figure 5.9. Average processing rates for speckle and checkerboard target patterns. ....	55
Figure 5.10. Lighting test: Sample horizontal displacement for speckle target pattern.....	56
Figure 5.11. Lighting test: Sample horizontal displacement for checkerboard target pattern.....	56
Figure 5.12. Natural feature test: Sample optical flow tracking of steel and concrete targets. ....	58
Figure 5.13. Natural feature test: Sample horizontal displacement for steel and concrete targets. ....	59
Figure 5.14. String pot comparison test: Sample optical flow tracking between initial and final image. ....	61
Figure 5.15. String pot comparison test: Sample string pot and OFM horizontal displacement..	62
Figure 5.16. Traffic signal structure test: Sample optical flow feature point selection. ....	63
Figure 5.17. Traffic signal structure test: Sample OFM horizontal and vertical displacement....	64
Figure 6.1. Camera mounted on bridge pier cap.....	68
Figure 6.2. Brattons Creek Bridge. ....	69
Figure 6.3. Pier cap study: Bridge instrumentation.....	70
Figure 6.4. Deflection due to pier cap rotation. ....	70
Figure A1. Traffic signal structure test: OFM displacement for test #1. ....	81
Figure A2. Traffic signal structure test: OFM displacement for test #2. ....	81
Figure A3. Traffic signal structure test: OFM displacement for test #3. ....	82
Figure A4. Traffic signal structure test: OFM displacement for test #4. ....	82
Figure A5. Traffic signal structure test: OFM displacement for test #5. ....	83
Figure A6. Traffic signal structure test: OFM displacement for test #6. ....	83

# **1 Introduction**

## **1.1 Background and Motivation**

With an average age of 43 years, many bridges in the U.S. will soon reach their design life and require rehabilitation or replacement (*Infrastructure Report Card 2017*). The state of the country's infrastructure, which extends beyond bridges, requires owners and transportation authorities to support diligent asset management and seek cost-effective methods for repairing or replacing such systems. Before large investments are made, regular assessment will occur to extend the design life as much as possible.

Inspection and assessment are necessary steps to maintain and ensure the safety and capacity of critical infrastructure systems such as bridges and other civil structures. Bridges, for example, are generally inspected every two years, as determined by the structure's age and condition. These inspections are typically visual, as structures that do not merit large concerns can be assessed well from the outside. For structures that are aging and not performing well in visual inspections, additional measures can be taken to assess their structural capacity using nondestructive evaluation and load testing. Traditional sensors, including accelerometers, strain gauges, and string potentiometers, can be difficult to place in remote, offshore, and complex environments where the structure is very tall or hard to reach. Vision-based systems offer an innovative approach for structural health monitoring because they do not require contact with the target structure. Innovative methods for sensing and evaluation can save time and money, while potentially providing more quantitative results than traditional subjective human inspections.

## **1.2 Research Objectives**

Previous work has confirmed the accuracy of a camera-based system using two-dimensional digital image correlation (2D-DIC) for post-processed bridge deflection measurement (Alipour et al. 2019). Experience from that work led to interest in a smart camera system capable of onboard image acquisition and processing. With the previous work as a baseline, this thesis research seeks to answer the following questions:

- (1) Can you use a vision-based system to monitor a structure that cannot be reached for attachment of traditional sensors?
- (2) Can you visually track a structure's natural features so that manmade target cards do not need to be mounted on the structure?

- (3) Can you develop a field deployable system that can be instrumented on the structure itself for monitoring when no fixed ground is available adjacent to the structure?
- (4) Can you create a vision-based system that can process images in the field quickly so that results can be reviewed before leaving the site?

Based on the questions proposed, the objective of this work was to develop a vision-based monitoring system that could collect contactless displacement measurements by tracking natural structural features with potential for real-time processing.

### **1.3 Thesis Outline**

This thesis explores the development of a contactless vision-based system for structural displacement measurement with real-time processing potential. Chapter 1 introduces the motivation and objectives for the work. Chapter 2 will provide an overview of relevant background information from the literature, including current vision-based structural health monitoring practices, image processing techniques, and computational methods. Chapter 3 will explore the specific methods to implement these objectives, including methods for image acquisition and algorithms for image processing. Chapter 4 will describe the final system, a series of lab tests and feasibility studies conducted to calibrate the parameters and validate the system under a variety of conditions, and field testing conducted to evaluate performance capabilities. Chapter 5 will present the results from the lab testing and field studies. Chapter 6 will draw conclusions from the results, discuss system limitations, and recommend steps for deployment of the system in the built environment.

## **2 Literature Review of Structural Health Monitoring**

This literature review will explore the current state of vision-based structural health monitoring (SHM) to present ample background surrounding the questions that this thesis seeks to answer regarding contactless displacement measurement, feature tracking, and real-time processing. Section 2.1 will describe traditional methods for SHM and their limitations. Section 2.2 will introduce vision-based SHM methods, review their applications, and provide some examples of their recent use. Section 2.3 will provide background on image processing methods that are typically used for such vision-based systems. Section 2.4 will provide background on computing methods in SHM and discuss the benefits of processing optimization.

### **2.1 Traditional Methods for Monitoring**

Structural systems such as bridges are monitored to study their in-service behavior. Monitoring systems can be designed for short term deployment, e.g. during controlled load tests, or long-term deployment to understand the behavior of a structure over time. With developments in electrical and computing technologies, monitoring methods are able to extend far beyond the capabilities of human-based visual inspection, which still remains standard for bridge inspections. Human-conducted visual inspection is very subjective to the user's vision, lighting conditions, and access (Hellier 2003). An inspector has the potential to miss small defects or any irregularities that are not visible on the surface-level, which may be critical. Visual inspection also lacks quantitative results, which makes measurement-based monitoring appealing.

Measurement-based monitoring of bridges, for example, typically occurs during ambient traffic or controlled load testing. Load testing provides a way to characterize the structural response of a bridge by quantifying the load distribution across the structure and its dynamic response. The results from load testing can be applied to load rating equations to predict if a structure is meeting or exceeding its capacity. Based on the result of load rating, a structure may be posted, so that only trucks up to a certain weight can drive across it, or be designated for further inspections ("Manual for Bridge Evaluation" 2018).

Traditional measurements typically found within SHM include but are not limited to acceleration, strain, tilt, temperature, and deflection. The main acquisition system for collecting accelerations is through accelerometers. Acceleration data is used to conduct a dynamic analysis of a structure, such as identifying natural frequencies, mode shapes, and damping ratios.

Acceleration measurements can also be collected using fiber-optics, as previous work has proved (Kim and Feng 2007). Additionally, smartphones are an innovative source for collecting deck acceleration measurements without requiring extensive wiring (Min et al. 2015). Strain gauges are the main source for strain measurements and are used in static or dynamic testing. Strain data from a collection of devices can be combined to understand the distribution of strain across a structure, which can be utilized in load rating procedures. Tiltmeters measure tilt, which is the rotation of a structure along an axis. Thermistors or thermocouples measure the thermal properties at specific points on a structure and are typically used alongside other measurement devices, such as strain gauges, to obtain a full-field structural response.

Deflection is displacement due to a load and deflection-based measurement acquisition typically occurs during static, quasi-static, or dynamic loading. Deflection distribution across a structure is another input option for load rating calculations. Some examples of deflection-based measurement systems include linear variable differential transducers (LVDTs), string potentiometers, GPS, lasers, and radar. LVDTs are highly accurate and can measure small scale displacements when fixed to a target structure (Ribeiro et al. 2014). String potentiometers can capture larger linear displacement measurements, but can be tedious to instrument and require a fixed ground position directly under the structure (Bartilson et al. 2015). GPS uses satellite connectivity to collect static displacements in real-time (Im et al. 2013). This poses difficulties when the system cannot maintain an unobstructed view with a satellite. Additionally, GPS devices can be expensive compared to other monitoring systems (Ribeiro et al. 2014). Laser vibrometers use laser interferometry to map the bridge response, but require a fixed ground point underneath the bridge (Nassif et al. 2005). Radar-based sensors are low-cost, small, and scalable, as well as suitable for short or long-term monitoring (Bartoli et al. 2008; Guan et al. 2019; Rice et al. 2012). However, they must be within 6 m of the target, or fixed onto the structure, in order to minimize error, which can limit their application in complex field conditions.

The methods described here either require placement on a fixed point on the structure or on ground within a prescribed distance from the structure. This can be limiting for structures in complex environments, such as offshore bridges over long waterways, bridges over traffic, or tall traffic structures, which are difficult to reach or have limited adjacent fixed ground locations. Deflection sensors are especially difficult to instrument in such scenarios. To solve these difficulties, vision-based systems offer an innovative solution for displacement measurement.

## 2.2 Introduction of Vision-based SHM Methods

Vision-based methods provide an innovative solution to some of the issues posed by traditional measurement systems because they do not require extensive wiring, do not require contact with the target structure (depending on the processing algorithms used), can be low-cost, and are usually easy to set up and operate (Feng and Feng 2018). They may be used to obtain displacement measurements from multiple points on a structure without needing to move the system. Additionally, vision-based SHM systems have many applications beyond displacement measurement.

### 2.2.1 Applications of Vision-based SHM

The science of photogrammetry is the cornerstone of vision-based monitoring systems. Photogrammetry is the use of photography to make measurements by taking the spatial coordinates of discrete points within the photograph (Jáuregui et al. 2003). Movement between photographs can be correlated to displacements, which can then be scaled into distances from calibration with known reference points. Displacement can be the final measurement based on the movement or the relationships can be applied to calculate strains within a region or to measure vibrations.

Vision-based monitoring systems have structural health monitoring (SHM) applications for: damage detection (Dworakowski et al. 2016; Kong and Li 2018; Zaurin et al. 2016); vibration monitoring (Busca et al. 2014; Caetano et al. 2011; Ji and Chang 2008; Xu et al. 2018); vehicle detection, classification, and tracking (Taylor and Lydon 2017); weigh-in-motion systems (Ojio et al. 2016); remote inspection via unmanned aerial vehicles (UAVs) (Brooks et al. 2018; Reagan et al. 2017; Yoon et al. 2018); load rating (Catbas et al. 2012; Peddle 2011); structural identification and model updating (Shafiei Dizaji et al. 2017); strain measurement (Pan et al. 2009a; b); and displacement measurement (Chiang et al. 2011; Feng et al. 2015a; Fukuda et al. 2013; Jáuregui et al. 2003; Pan et al. 2016; Peddle et al. 2011; Ribeiro et al. 2014; Shariati and Schumacher 2017; Xu et al. 2019; Ye et al. 2013; Yoneyama and Ueda 2012). The application of vision-based SHM for displacement measurement will be explored next.

### 2.2.2 *Current Vision-based Displacement Systems*

The use of vision-based monitoring for displacement measurement constitutes the focus of this research. Most vision-based displacement measurement systems use global methods by featuring the camera placed on fixed ground a set distance away from the point of interest on the structure. For example, the Feng group measures dynamic displacement of bridges using a camcorder running at 10 fps on a tripod on an adjacent ground reference location 300 m away (Feng and Feng 2017). Similarly, the Pan group performed real-time measurement of global vertical deflections of a railway bridge that reported subpixel accuracy and included calibration for off-axis camera alignment (Pan et al. 2016). Some parameters to account for when designing vision-based systems include: availability of space to position camera near the structure; perpendicularity of camera to target (Pan et al. 2016); lighting availability (Lee et al. 2017; Luo et al. 2018); camera movement (Xu et al. 2019; Yoneyama and Ueda 2012); desired computation speed, etc. The methods for image processing of such vision-based systems will be reviewed next.

## 2.3 **Background on Image Processing Methods**

The use of vision-based systems presents three main processing-related tradeoffs: the need for physical targets vs. contactless acquisition; computational speed vs. accuracy; and overall computational cost. These factors are all accounted for based on the robustness of the algorithm used. Image processing algorithms can be divided into two main fields: area-based template matching and feature-based template matching. For both approaches, template matching forms the basis for tracking through which an image is processed by comparing a predefined region of interest (ROI) of the image to that of a source image. A pixel by pixel comparison between the template and source occurs. Scale calibration then allows for a matching of the pixel result with a physical displacement value. This constitutes the ability to track an object between consecutive images or video frames.

### 2.3.1 *Area-based Template Matching*

Area-based template matching includes the installation of a target card or panel on the structure being monitored and follows changes in area on the card, which can be correlated with movement of the structure. With the use of a card comes questions of card pattern, size, etc., which are developed based on the requirements of the algorithm utilized. Target patterns that have been

used previously include speckle dots (Alipour et al. 2019), solid black with four white circles (Fukuda et al. 2010; Ribeiro et al. 2014), QR code-type patterns (Feng and Feng 2017; Shariati and Schumacher 2017), and a bullseye target with a plus sign at the center (Jáuregui et al. 2003). Some examples of area-based algorithms include digital image correlation (DIC) (Pan et al. 2013; Shafiei Dizaji et al. 2017; Yoneyama et al. 2007), orientation code matching (OCM) (Feng and Feng 2017; Feng et al. 2015b; Fukuda et al. 2013; Ullah et al. 2001), upsampled cross correlation (UCC) (Feng et al. 2015a; Guizar-Sicairos et al. 2008), and the use of virtual visual sensors (VVS) (Shariati and Schumacher 2017; Song et al. 2014). Area-based techniques such as DIC offer the ability to use full-field deformation data to calculate strain measurements (Baker and Matthews 2001; Bruck et al. 1989; Pan et al. 2009b; Sutton et al. 2009).

### 2.3.2 *Feature-based Template Matching*

Feature-based template matching detects and tracks local features on the object of interest (such as bolts, cracks, edges of structural members, etc. (see Figure 2.1)) and thus does not require installation of a physical target card. Some examples of feature-based algorithms include Kanade-Lucas-Tomasi (KLT) tracking (Kong and Li 2018; Lucas and Kanade 1981; Morlier and Michon 2010; Taylor and Lydon 2017; Yoon et al. 2016) and the optical flow method (Caetano et al. 2011; Horn and Shunk 1981; Ji and Chang 2008; Lim et al. 2005). Part of the process of feature tracking requires the detection of features to track, which typically take the form of key-points, blobs, or corners. Some algorithms that facilitate feature detection include Harris corner detection (Harris and Stephens 1988), Shi-Tomasi feature detection (Shi and Tomasi 1994), scale-invariant feature transform (SIFT) (Lowe 2004), features from accelerated segment test (FAST) (Rosten and Drummond 2006), speeded-up robust features (SURF) (Bay et al. 2008), binary robust independent elementary features (BRIEF) (Calonder et al. 2010), binary robust invariant scalable keypoints (BRISK) (Leutenegger et al. 2011), and fast retina key-point (FREAK) (Alahi et al. 2012). The feature-based tracking methods listed have commonly been applied to vibration measurement (of cables or other structural components), crack detection, and displacement measurement.



Figure 2.1. Trackable structural features.

## 2.4 Background on Computing Methods in SHM

### 2.4.1 Real-time vs. Post-processing

The photographs obtained during field testing must be processed using algorithms from Section 2.3. Depending on the design of the system and robustness of the algorithm, this processing is either completed in real-time while the testing is conducted or post-processed afterwards. Based on the quantity of images collected over the duration of the testing, post-processing can take hours, if not days, in order to analyze all of the images. Because of this computational cost, real-time processing can be very desirable. Real-time processing is achieved when the total time for computation is less than the sampling rate of the system (Torbol 2014). The higher the sampling rate is, the harder it will be for the system to process the data as it comes in.

The opportunity for real-time computing of vision-based measurements is limited by the ability to connect the camera to a computer. In laboratory environments, this is typically the general setup. However, the application of real-time processing in the field is more difficult. With most off-shore applications (e.g. where the camera is located on fixed ground), this can be done, and thus a relatively standard computer can process the images as they come in (Pan et al. 2016). For more remote acquisition, this is only possible if the camera is connected wirelessly to a computer or if a “smart camera” is employed. While the technology used for real-time processing is commonplace for UAVs and autonomous vehicles (Betke et al. 2000; Brooks et al. 2018; Reagan et al. 2017; Yoon et al. 2018), the application of it to the field of civil engineering for static purposes is highly underutilized. As the use of real-time image processing for remote acquisition has not been explored in depth in SHM literature, the potential for real-time processing will serve as motivation throughout this study.

#### 2.4.2 Introduction of GPUs for Computing

Data processing is primarily limited by the ability of the processing unit used. The growth of graphics processing units (GPUs) for gaming systems has contributed to the development of computational power with research applications. Instead of sending data through a central processing unit (CPU), it can be sent to a GPU, which is made up of hundreds or thousands of cores, compared to a CPUs' 2 to 16 cores (Torbol 2014). Thus, this allows for faster processing, given that a code can be parallelizable. GPUs have developed significantly over the past two decades due to advances in circuitry. An illustration of the increase in memory bandwidth of GPUs versus CPUs over a thirteen-year period is shown in Figure 2.2. As can be seen, the processing power of two NVIDIA GPUs (GeForce and Tesla) has increased dramatically compared to that of the Intel CPU given.

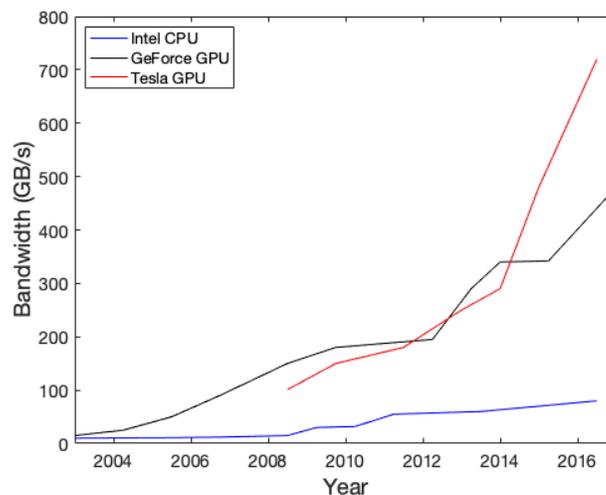


Figure 2.2. CPU vs. GPU bandwidth trends from 2003 to present (adapted from NVIDIA Corporation 2018).

GPUs are common for high-level computing especially for image processing applications where the graphics can be processed more efficiently than by a CPU (Almeida 2015; Asano et al. 2009). The strength of GPUs lies in the opportunity for parallelization. Parallelization allows different parts to be individualized with iteration through the different GPU cores. One work using parallelization for normalized cross correlation computation of ultrasound strain images through Matlab, a CPU, and a GPU found that the GPU reduced the computation time by a factor of 130

(templates of pixel size 256 x 9) to 376 (16 x 9 pixels) over the Matlab processing (Idzenga et al. 2014).

In the field of civil engineering, GPUs and code parallelization have been utilized for post-processing for several years (Adeli and Kamal 1993). One study applied parallel processing techniques to enhance the speed of finite element analysis, which is notably computationally costly when large matrices are used (Sotelino 2003). Another work collected images of a bridge cable during field testing and then post-processed those images on a laboratory GPU using parallel computing (Park et al. 2018), while another study used a GPU to reduce the time of modal analysis computations using a parallel frequency domain decomposition algorithm (Torbol 2014). With these notable advantages, there are also some constraints presented by the use of GPUs when it comes to smooth integration into real-time systems (Elliott and Anderson 2011). Overall, the use of GPUs for real-time computing has proven to be a valuable computational tool and will be considered in the development of this work.

## **3 Experimental Methods**

### **3.1 Motivation**

The goal of this work was to develop a field deployable, contactless vision-based monitoring system that can acquire and quickly process images for localized displacement measurements of civil engineering structures. Vision-based systems offer an innovative approach to structural health monitoring because they do not require contact with the target structure in the way that traditional sensors, such as accelerometers, strain gauges, and string potentiometers, do. This chapter will explore the methods for image acquisition and algorithms for image processing to develop the system. Section 3.2 will browse computing devices and camera and lens pairs. Section 3.3 will introduce available software and describe two relevant area and feature-based tracking algorithms.

### **3.2 Image Acquisition**

#### *3.2.1 Computing Device*

To support onboard processing and manipulation of images, a computing device was necessary to control the camera to be used in this computer vision-based system. Essentially, this enables control of the system on the level of a “smart” camera. In previous work, all image processing was performed after the completion of the testing back in the lab on desktop computers (Alipour et al. 2019). This process, specific to the image processing algorithms used, proved to be time consuming. Additionally, there was no way to check the quality of the obtained images before departing the field site. Thus, investigation into the use of a field deployable computing device increases the capability of this system to obtain measurement results while still at the field site.

A basic smart camera that was explored is the Raspberry Pi. Raspberry Pi is a credit card-sized single board computer that offers a moderate level of computational capacity. The Raspberry Pi 3 is shown in Figure 3.1 and the specifications of this model evaluated are shown in Table 3.1. One benefit of the Raspberry Pi system is the connectivity offered by such devices, which creates the ability to sync a collection of the devices together to create a network of sensors. The Raspberry Pi initially served as a small-scale exploration into the desired type of computing system for civil engineering field applications; however, it was not chosen for use in this work. The main limitations of this system were due to the lack of remote connectivity (requires WiFi or ethernet) and the computational capacity of the onboard processing unit. A processing unit that had a

capacity for more complex computing was desired to enable work with large quantities of high-resolution images (requiring a high acquisition rate) and processing through a system architecture with potential for real-time processing. These were deemed system requirements based on inefficiencies in the previous work.



Figure 3.1. Raspberry Pi 3 Model B.

Table 3.1. Raspberry Pi 3 specifications.

<b>Component</b>	<b>Specification</b>
Model	Raspberry Pi 3 Model B
CPU	ARM v8 quad core (64-bit)
CPU Speed	1.2 GHz
Memory	1 GB SDRAM @ 400 MHz

Beyond the Raspberry Pi, other single board computers on the market included ODROID, Arndale, and NVIDIA products. After preliminary review of the capacities of these systems, their applications by other researchers, and the depth of documentation for their use, NVIDIA was chosen as the main product line to consider.

NVIDIA computing devices are popular due to their versatility, GPU processing capabilities, functionality of the CUDA interface, and cost effectiveness. NVIDIA products are frequently upgraded and a large user community exists through an online forum for developers. The computational capabilities of NVIDIA devices have been studied in the literature (Amert et al. 2017; Otterness et al. 2017; Torbol 2014). NVIDIA’s Jetson line of computing devices is designed for applications in embedded computing, particularly for autonomous systems requiring real-time image processing and machine learning, and was selected for this study.

The Jetson TK1, the first edition of NVIDIA’s Jetson modules, was released in 2014 with the computing power equivalence of four Raspberry Pi’s, all on a single chip (Wolfer 2015). The most recent Jetson module, the Jetson TX2, was released in 2017, while its predecessor the TX1 was released in 2015 (Smith 2017). The Jetson TX2 developer kit (the credit card-sized computer attached to a carrier board) is shown in Figure 3.2 (TX1 looks similar) and a comparison of the main specifications between the TX1 and TX2 is presented in Table 3.2.

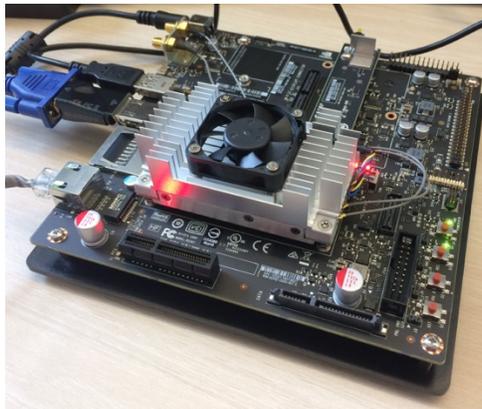


Figure 3.2. NVIDIA Jetson TX2.

Table 3.2. Comparison of specifications between NVIDIA Jetson TX1 and TX2.

<b>Component</b>	<b>Jetson TX1</b>	<b>Jetson TX2</b>
GPU	NVIDIA Maxwell	NVIDIA Pascal
CPU	Quad ARM A57/2 MB L2	HMP Dual Denver 2/2 MB L2 + Quad ARM A57/2 MB L2
Video	4K x 2K 30 Hz Encode (HEVC) 4K x 2K 60 Hz Decode (10-Bit Support)	4K x 2K 60 Hz Encode (HEVC) 4K x 2K 60 Hz Decode (12-Bit Support)
Memory	4 GB 64-bit LPDDR4 25.6 GB/s	8 GB 128-bit LPDDR4 59.7 GB/s
CSI Connection Speed	1.5 Gbps/Lane	2.5 Gbps/Lane
Data Storage	16 GB	32 GB
Operating Life	5 years	

As the TX2 is a generation after the TX1, thus serving as an upgraded version of its abilities, it exceeds and practically doubles the computing power of the TX1. All of the specifications listed in Table 3.2 were considered relevant to evaluate for the purposes of the proposed vision-based work. At the time of this research, the two modules were on the market for

the same price, so the newer Jetson TX2 and its development kit were chosen for creation of the vision-based system. NVIDIA also produces a version of the TX2, called the Jetson TX2i, which features similar processing capabilities but a more rugged design and a longer operating life (10 years) for use in industrial environments (NVIDIA Corporation 2019). Based on the goals of this work, the TX2i was deemed unnecessary for this initial exploration.

The Jetson TX2 features a Linux-based operating system running Linux for Tegra (specific version at time of flash was L4T 28.2.1). The software package that can be flashed onto the system is called JetPack. JetPack 3.3 was installed on the TX2 with the following components: CUDA toolkit for the host (Ubuntu) and target platform, the latest NVIDIA Developer Tools (Tegra Graphics Debugger 2.5 and NVIDIA System Profiler 4.0), VisionWorks 1.6, cuDNN v7.1.5, Multimedia API v28.2, OpenCV 3.3.1, and TensorRT 4.0 GA. Some of the relevant physical components contained in the 17 x 17 cm Jetson TX2 developer kit include: DC power port, integrated antennas, micro USB port, USB Type A port, HDMI port, ethernet port, and a camera module. Ultimately, the Jetson TX2 promised to meet the needs of this work including high-quality image acquisition and processing with potential for field deployment.

### 3.2.2 Camera and Lens

Next, a camera needed to be selected for use in this smart camera system based on optical specifications and compatibility with the chosen computing device. An appropriate lens must be paired with the camera in order to meet the desired magnification requirements. One measure of applicability of a camera and lens pair is checking that the field of view is within the desired range of testing based on the internal imaging system parameters. Equation 3.1 provides the field of view (FOV) proportional relationship, where  $w$  is the width of the camera sensor,  $W$  is the width of the FOV,  $f$  is the lens focal length, and  $d$  is the distance to the target. To calculate the height  $H$  of the FOV, the heights  $h$  and  $H$  can be substituted for the widths  $w$  and  $W$  in Equation 3.1, respectively. Thus, given a sensor size  $w \times h$ , lens focal length  $f$ , and a distance  $d$ , the minimum FOV dimensions  $W \times H$  can be determined. Conversely, required sensor size or lens focal length can be determined based on a desired FOV size. The relationship between these parameters can be visualized in Figure 3.3(a). To determine an effective target size ( $t$ ) within the given image (assuming a square target of size  $t \times t$ ), a nominal margin ( $m$ ) around the target can be considered, as represented in Equation 3.2 and shown in Figure 3.3(b). The required focal length of the lens ( $f_{req}$ ) can be

determined by rearranging Equation 3.2 into Equation 3.1, as shown by Equation 3.3. By further rearranging Equation 3.3, the minimum target size ( $t_{min}$ ), based on the maximum focal length of a known lens ( $f_{max}$ ), can be determined through Equation 3.4.

$$\frac{w}{W} = \frac{f}{d} \rightarrow FOV (W) = w \times \frac{d}{f} \quad (3.1)$$

$$FOV (W) \geq t + m \quad (3.2)$$

$$f_{req} \leq w \times \frac{d}{t + m} \quad (3.3)$$

$$t_{min} \geq w \times \frac{d}{f_{max}} - m \quad (3.4)$$

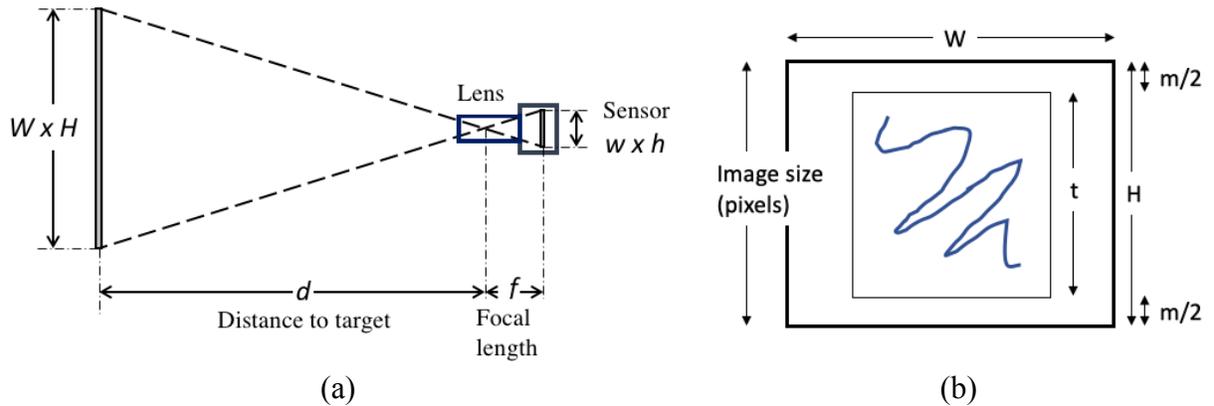


Figure 3.3. (a) Field of view estimation and (b) Image dimensions.

Previous testing was conducted at a distance of 37.5 ft, which represents half of a bridge with a 75 ft span (Alipour et al. 2019). For this work, a representative distance of 30 ft was used in subsequent calculations for reference. For target size, a 6 x 6 in. card and a nominal margin of 0.5 in. was used. The 6 x 6 in. card was chosen because that was the card size used previously for bridge field testing as it fits under girders, provides enough surface area for adhesion, and works within the imaging distance and desired magnification level (Alipour et al. 2019). The nominal margin represents a comfortable window around this. This target card size and margin led to a minimum FOV size of 6.5 x 6.5 in. Therefore, the ideal camera and lens for this study would meet these geometric requirements of being capable of viewing a 6 x 6 in. target with a 0.5 in. margin from at least 30 ft away.

As introduced in Section 3.2.1, the Raspberry Pi was explored as a smart camera system. During its investigation, a camera module supported by a pan-and-tilt mechanism was added to the system and offered basic functionality (see Figure 3.4). The specifications for the camera module are presented in Table 3.3. The camera sensor was a Sony IMX219. Using Equation 3.1 and the given system specifications, the minimum field of view at 30 ft was 435.1 x 326.8 in. Thus, this camera was deemed inadequate for the distances and magnification required by this work, which served as another reason for why the Raspberry Pi system was not continued with.

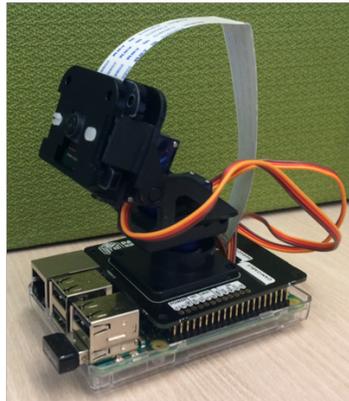


Figure 3.4. Raspberry Pi camera module with pan-and-tilt mechanism.

Table 3.3. Raspberry Pi camera module specifications.

<b>Component</b>	<b>Specification</b>
Camera Model	Raspberry Pi Camera Module v2
Camera Sensor	1/4" (3.674 x 2.760 mm)
Camera Resolution	3280 x 2464 pixels
Lens Focal Length	3.04 mm

The Jetson TX2 includes an onboard camera module and the specifications for it are provided in Table 3.4. This camera is capable of taking high resolution images up to 5 MP and can be accessed through a GStreamer pipeline. However, increasing the resolution of the images decreases the acquisition frame rate. The sensor size is the same as the Raspberry Pi camera module, so it was also deemed incapable of providing the depth of field required for this work.

Table 3.4. NVIDIA Jetson TX2 camera module specifications.

<b>Component</b>	<b>Specification</b>
Camera Model	Jetson TX2 Camera Module
Camera Sensor	1/4" (3.674 x 2.760 mm)
Camera Resolution	2560 x 1920 pixels
Lens Focal Length	Fixed Focus

The main requirements for a camera and lens combination were that it meets the field of view required dimensions and can connect to the TX2. External camera types available for use and supported by the TX2 include CSI (camera serial interface), USB 2.0 (webcams), and USB 3.0 cameras. Previous vision-based work (lab study and field application) used the GoPro Hero 3+ camera with a Tamron A17 70-300 mm lens for deflection sensing (Alipour et al. 2019). The GoPro Hero 3+ is shown in Figure 3.5(a) and the specifications are provided in Table 3.5. This camera offered WiFi connectivity, image and video acquisition capabilities, and high-resolution images (single image up to 10 MP). The Hero 3+ has two relevant modes of image collection: time-lapsed imagery and high-definition video. Previous work evaluated both modes and determined that time-lapse images brought limitations when it came to image acquisition rate and video brought limitations with video compression. For this work, the Hero 3+ was considered with the Tamron Model A17 70-300 mm lens, which is shown in Figure 3.5(b) and whose specifications are provided in Table 3.6. The lens aperture was modified to f/8 to adjust the exposure for low-light scenarios, such as a dark testing room or evening field test. From a distance of 30 ft at maximum focal length (300 mm), the Hero 3+ can view a minimum target of size 4.8 x 6.4 in. This field of view is within the requirements of this work. However, the Hero 3+ was less desirable in terms of other criteria. Previous testing showed that sometimes the WiFi connectivity delayed or disappeared completely during testing. Outside of WiFi, the Hero 3+ features an HDMI connection and thus requires a connector module to interface physically with the TX2. Thus, a camera that did not run over WiFi strictly and that could connect with the Jetson TX2 was desired. A camera that could interface with the TX2 in such a way that could enable the potential for processing in the field was deemed important.



(a)



(b)

Figure 3.5. (a) GoPro Hero 3+ and (b) Tamron A17.

Table 3.5. GoPro Hero 3+ camera specifications.

Component	Specification
Camera Model	GoPro Hero 3+ Silver
Camera Sensor	1/2.7" CMOS (4.04 x 5.36 mm)
Camera Resolution	Single image: 3680 x 2760 pixels Video: 1920 x 1080 pixels

Table 3.6. Tamron A17 lens specifications.

Component	Specification
Lens Model	Tamron Di LD Macro Model A17
Lens Focal Length	70-300 mm
Lens Aperture	f/4-5.6 (modified to f/8)

Some cameras with direct application to the Jetson are manufactured by e-con Systems, FLIR, and Leopard Imaging. Ideally, a camera that did not require a connector module would be chosen in order to minimize the programming requirements. The TX2 features a USB3 connection, so the rest of this study explored cameras that include USB3 connectivity. As some of these cameras can be very expensive, low-cost options were primarily considered for this feasibility study. FLIR (formerly Point Grey) was the main manufacturer evaluated.

FLIR's Grasshopper 1394b (Model GRAS-50S5M-C) camera was studied. The sensor on this camera is a Sony ICX625 with 2/3" dimensions and it is capable of 5.0 MP image resolution. However, it features Firewire connectivity and is unable to connect to the TX2 directly, which was not favorable. Additionally, the FOV at 30 ft with a 300 mm lens was 10.6 x 7.9 in. The next camera studied was FLIR's Grasshopper3 (Model GS3-U3-23S6M). The Grasshopper3 (GS3) has the ability to interface with ARM devices like the TX2 and can run through a USB3 connection, so it was able to connect directly to the TX2. The GS3 is shown in Figure 3.6(a) and specifications

of the specific model of GS3 are provided in Table 3.7. Like the Hero 3+, it has a CMOS image sensor, but it is monochrome so it only produces gray scale images. The sensor is a Sony IMX174, which was larger than the Hero 3+ sensor. The GS3 was paired with the Tamron Model 672D 75-300 mm lens, which is shown in Figure 3.6(b) and whose specifications are provided in Table 3.8. This lens features an adjustable aperture setting with seven exposure levels: f/4 (wide aperture, brightest exposure), f/5.6, f/8, f/11, f/16, f/22, and f/32 (small aperture, darkest exposure). The adjustable aperture allows for manual control of the lens exposure, which may need to be manipulated at times based on the given lighting conditions during use and the desired depth of field. For example, brighter exposure levels can be used in low-light scenarios to increase image visibility, while the darker exposure levels can be used in brighter scenarios to balance the image. From 30 ft and at maximum focal length of 300 mm, the FOV of this system was 13.6 x 8.5 in. This did not meet the previously stated requirement of 6.5 x 6.5 in. However, the TX2 connectivity offered by this camera was desirable and available, so the FOV parameters were adjusted to accept this camera and lens combination as the primary optical equipment for use with the Jetson. If a larger margin of 8 in. around the target was allowed, the total image size becomes 14 x 14 in., which can be seen by the system at a focal length of 290.6 mm (covered by the 300 mm lens). The consequences of increasing the field of view will be discussed in the context of the experimental setup in the next chapter.

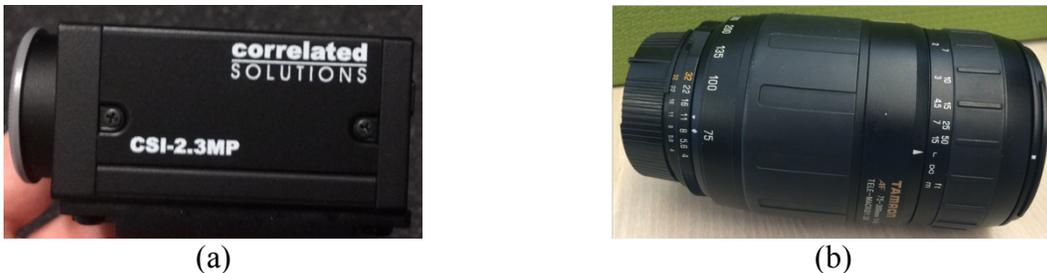


Figure 3.6. (a) FLIR Grasshopper3 and (b) Tamron 672D.

Table 3.7. FLIR Grasshopper3 camera specifications.

Component	Specification
Camera Model	Grasshopper3 (GS3-U3-23S6M)
Camera Sensor	1/1.2" CMOS (11.3 x 7.1 mm)
Camera Resolution	1920 x 1200 pixels
Camera Firmware	2.30.3.0

Table 3.8. Tamron 672D lens specifications.

<b>Component</b>	<b>Specification</b>
Lens Model	Tamron Macro Model 672D
Lens Focal Length	75-300 mm
Lens Aperture	f/4-5.6

### 3.3 Image Processing for Displacement Measurement

#### 3.3.1 Open-source and Commercial Software

The goal of this work was to create a vision-based system for displacement measurements. Thus, in addition to image acquisition using the camera system as controlled by the computing device, the images need to be processed to provide displacement measurements. Some area and feature-based template matching algorithms that have previously been applied to monitoring of civil engineering structures were presented in Section 2.3. Using these known methods as a foundation, research was conducted to locate open-source and commercial software applying these methods. Several examples of area and feature-based template matching software packages are listed in Table 3.9 and Table 3.10. The commercial software VIC-2D from Correlated Solutions was used previously (Alipour et al. 2019). This digital image correlation-based package proved accurate, but required hours to process large sets of high-resolution images. For this main reason, other software packages were studied.

Table 3.9. Commercial software for image processing.

<b>Software</b>	<b>Owner</b>	<b>Function</b>
ARAMIS	Trilion/gom	3D-DIC displacement, strain
DaVIS	LaVision	3D-DIC deformation, strain
VIC-2D	Correlated Solutions	2D-DIC displacement, strain
VIC-3D	Correlated Solutions	3D-DIC displacement, strain

Table 3.10. Open-source software for image processing.

<b>Software</b>	<b>Platform</b>	<b>Function</b>
CR Toolbox (Barbacci et al. 2014)	Matlab	Compute optical flow and feature tracking from video
Digital Image Correlation Engine (DICE) (Turner 2015)	Custom	Find displacement and strain for material characterization
Digital Image Correlation and Tracking (DICT) (Eberl et al. 2010)	Matlab	Find displacement and strain in images
Efficient Subpixel Image Registration by Cross-Correlation (ESR) (Guizar-Sicairos et al. 2008)	Matlab	Image registration
Fast Iterative Digital Image Correlation (FIDIC) (Bar-Kochba et al. 2015)	Matlab	Find displacements in 2D images
Feature Tracker (Dilshan 2017)	Matlab	Track features using KLT algorithm
Ncorr (Blaber et al. 2015)	Matlab	Find displacement and strain in images
OpenCV (Intel Corporation 2019)	Custom	Computer vision and machine learning software library
Yet Another Digital Image Correlation Software (YADICS) (Seghir et al. 2014)	Custom	Find displacement and strain in images for 2D fluid and solid and 3D tomographic applications

As commercial software had proven to be accurate, it was assumed that the other commercial packages listed in Table 3.9 would provide, essentially, similar capabilities to VIC-2D. Open-source software is usually updated frequently due to the nature of online collaboration and, unlike commercial software, available at little to no cost. However, open-source software can be very specific to the creator’s purpose, making it less versatile for other applications, and may be less user friendly and require substantial background knowledge in image processing for proper use. Of the open-source software presented in Table 3.10, all are Matlab-based except for DICE, OpenCV, and YADICS. Most of the open-source software presented are built around the use of correlation and are focused on material characterization, making them more robust than needed for rigid body displacement measurements. Because of this and because correlation had been used previously, feature tracking methods were of primary interest.

The feature-based software presented had similarities. CR Toolbox uses optical flow and Feature Tracker uses Kanade-Lucas-Tomasi tracking methods. Both of these methods appeared to be promising for the rigid body displacement measurement goals of this work. Building off of these methods, it was found that the OpenCV library was accessible on the TX2 through C++ and Python. The OpenCV library provides many different image processing functions and capabilities.

This led to the decision to use the KLT-based optical flow method through OpenCV functions. Throughout the rest of this work, KLT-based optical flow will serve as the primary image processing method and correlation will be used in order to validate it. The specific open-source software studied here will be referenced in the development of algorithms that meet the exact image processing requirements of this work.

### 3.3.2 Normalized Cross-Correlation

Correlation is an area-based template matching technique that compares the similarities between images. Its methods can be applied to both 2D and 3D measurements and this work will focus on 2D. Normalized cross-correlation is a conservative form of correlation that can be used for image processing. From a template image, a region of interest (ROI) is chosen that is searched for over subsequent images using prescribed step and subset sizes. Chosen step and subset sizes primarily affect the processing speed of images and the spatial resolution of the results. Through this act of template matching, a correlation between images can be enumerated which corresponds to the difference between them. The normalized cross-correlation calculates the correlation coefficient between a template image and a new image, which is in the range of  $[-1,1]$  where 1 indicates perfect correlation (100% match), 0 indicates no correlation (0% match), and -1 indicates perfect anti-correlation (100% opposite match). Between identical images, the correlation would be 100%.

The method of normalized cross-correlation to obtain correlation coefficients between two images has three steps: (1) calculation of cross-correlation between the two images; (2) calculation of the local sums; and (3) use of local sums to normalize the cross-correlation (Lewis 1995). These steps are all combined within Equation 3.5, which computes the correlation coefficient  $C$  in terms of  $x$  and  $y$ , where  $T$  is the template image,  $I$  is the new image,  $\mu$  is the mean,  $\sigma$  is the standard deviation, and  $n$  is the number of pixels in the image. Because Equation 3.5 includes subtraction of the means, it is called zero-normalized cross-correlation. Removing the local mean subtraction would create the normalized cross-correlation equation. In this work, zero-normalized cross-correlation was conducted using the Matlab function *normxcorr2* (The MathWorks, Inc. 2019).

$$C(x, y) = \frac{1}{n} \sum_{x,y} \frac{(I(x, y) - \mu_I)(T(x, y) - \mu_T)}{\sigma_I \sigma_T} \quad (3.5)$$

In these applications, the peak correlation in  $x$  and  $y$  planes can be used to calculate the displacement in  $x$  and  $y$  between two images. The difference in location of the peak between two images is the displacement between them. This displacement is in pixel coordinates and can be scaled to physical units using a known distance in the images as a scale factor. The displacement can be calculated for each of the subsets, but overall for a rigid body acting uniformly, the displacement is the same for the entire object. When the displacement is known over a region, the deformation between two points can be quantified and this can be used to construct a strain field for the object. The described template matching between images within the broader field of digital image correlation (DIC) is illustrated in Figure 3.7.

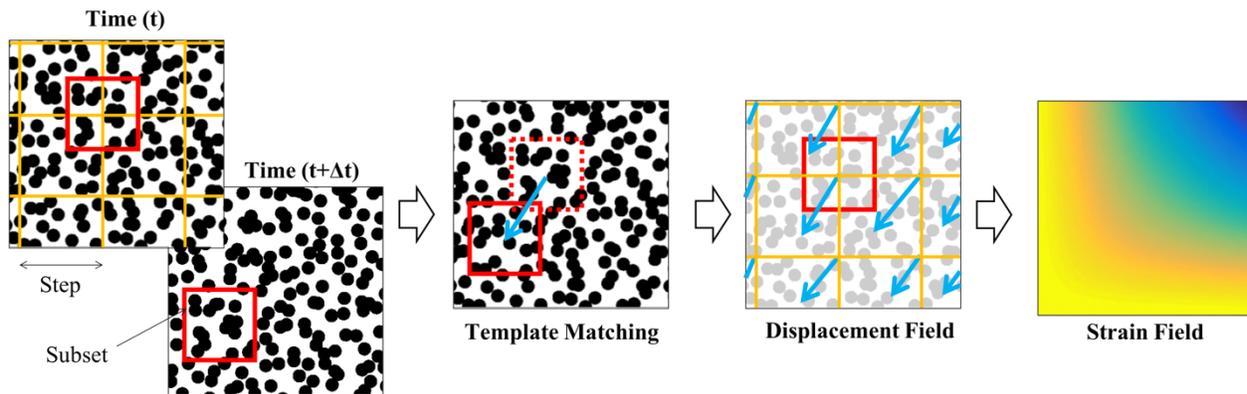


Figure 3.7. Digital image correlation process.

One of the required conditions for correlation includes the use of a patterned target. Traditionally, this is a high-contrast random speckle pattern of a particular speckle size and spacing based on the desired resolution of the system. The pattern can be applied manually to the object (drawing points with a marker) or feature the use of a printed card to be placed on the object. The size of the speckles is determined by the desired spatial resolution of the results. Minimum speckle size can be calculated using Equation 3.6 and 3.7. Equation 3.6 exhibits the relationship between the dimensions of the image and the object, both in pixel size and physical length (millimeters). Equation 3.7 calculates the size of the objects (speckle dots) ( $S_m$ ) using the field of view ( $FOV$ ), image width (or height) in pixels ( $W_p$ ), and object size in pixels ( $S_p$ ). Previous work used a speckle size of 2 mm for a 6.5 x 9.5 in.  $FOV$  and an image resolution of 1080 x 1920 pixels (2 MP) given a desired speckle pattern of 15 pixels (Alipour et al. 2019).

$$\frac{\text{image size (pixel)}}{FOV (mm)} = \frac{\text{object (pixel)}}{\text{object (mm)}} \quad (3.6)$$

$$S_m = \frac{FOV}{W_p} S_p \quad (3.7)$$

The advantages of using correlation for image processing include the quality of processing and the ability to calculate strain. High resolution images can lead to a fine representation of displacement with a high accuracy. For example, noted resolutions of DIC specific applications include 0.03 mm (Peddle et al. 2011) and 0.05 pixels (Pan et al. 2016).

Disadvantages of this correlation include the need to use a target pattern, which limits the ability to perform fully contactless measurements, as the artificial target must be installed on the target structure. For this 2D estimation, the deformation or displacement of the object must be in-plane. Additionally, the algorithm can be computationally expensive because each image is evaluated as a series of subsets based on the given ROI. This leads to high quality processing, but at a loss of speed of the computations. 2D digital image correlation was used previously with high accuracy (0.2 mm [0.0075 in.]) through the commercial software VIC-2D, but proved to be computationally costly, which is why other processing alternatives are being considered in this study (Alipour et al. 2019).

### 3.3.3 *Optical Flow Method*

Optical flow is the estimated motion of an object or image through differential tracking. Figure 3.8 illustrates the estimated flow based on the motion of an object over time. The connecting arrow represents the displacement vector of the object over five consecutive points of motion. One method of solving for optical flow is the Lucas-Kanade method. This feature-based method of template matching assumes constant flow and solves for the flow using least squares (Lucas and Kanade 1981). The Lucas-Kanade-based optical flow method (OFM) has three steps: (1) feature selection; (2) feature tracking; and (3) displacement measurement.

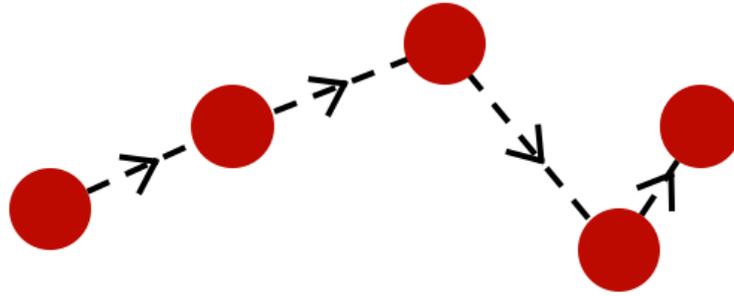


Figure 3.8. Optical flow estimation of an object through space.

To select features, Shi-Tomasi feature point detection is implemented to decide on features that are “good” for tracking (Shi and Tomasi 1994). They define a “good” feature as one that can be tracked easily by the tracker because the feature itself optimizes the quality of the tracker’s tracking. Such features may be defined by dissimilarity with the surrounding region or having high texture content. This detection and selection are important steps to improve the overall tracking accuracy. The Shi-Tomasi feature detection process is based on defining the dissimilarity between two points as a “corner.” To do this, a threshold value must be provided and if a pixel is above the threshold value, it is “selected” as a corner. This corner selection is based on Equation 3.8, where  $\lambda_1$  and  $\lambda_2$  are the eigenvalues from the image matrix and  $\lambda_t$  is the predefined threshold. Figure 3.9 illustrates the thresholding criteria where  $\lambda_{\min}$  is the predefined threshold set as a minimum. When the eigenvalues are above the threshold, they are considered corners (green region). If one eigenvalue passes but the other does not, the point is defined as an edge (gray region). If both eigenvalues are lower than the threshold, they are rejected as corners (red region). Thus, an entire image is scanned and, based on the given criteria, a set of corner points that fit within the given threshold values are determined. The threshold can be highly selective or more inclusive based on the amount of contrast expected in the images to be evaluated and the desired number and quality of feature points. The Shi-Tomasi method is similar to Harris corner detection, which uses a slightly different formulation for calculating the minimum of the two eigenvalues (Harris and Stephens 1988).

$$\min(\lambda_1, \lambda_2) > \lambda_t \quad (3.8)$$

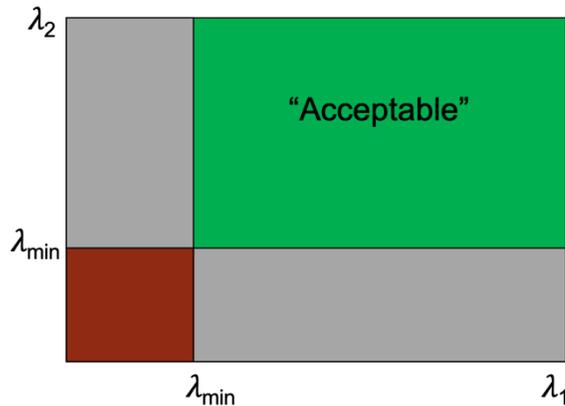


Figure 3.9. Shi-Tomasi corner detection using threshold values (adapted from Intel Corporation 2019).

For this feature point detection, the OpenCV function *goodFeaturesToTrack* was used (Intel Corporation 2019). The parameters are: number of corners, quality level, minimum distance, and block size. The number of corners limits the maximum number of corners to be returned. Quality level is a value between 0 and 1 which is multiplied by the quality measure of the best corner to create a minimum quality measure. Corners below this measure are rejected. For example, if the quality measure of the best corner is 1000 and the quality level is set to 0.9, all detected corners with a quality measure below 900 will be rejected (this example represents a highly selective scenario). Alternatively, a lower quality level will result in a larger quantity of corners to be retained with the natural tradeoff of these corners being of lower quality. The next parameter is minimum distance, which is the minimum Euclidean distance between the corners detected to prevent bunching up of feature points. This takes the best corner from the sorted list of corners after the quality check and removes any corners within that minimum distance around it, then evaluates the next best corner. Block size creates a square block of the given size around each pixel in the image which is used to solve for the eigenvalues  $\lambda_1$  and  $\lambda_2$  for each block of that size within the image. After these last three parameters have been evaluated in the given image, a series of corners no greater than the maximum number of corners will be returned.

After features have been selected, they are tracked using the estimation of optical flow. The differential optical flow equation is used to compute the optical flow vector where  $I_x(q_i)$ ,  $I_y(q_i)$ , and  $I_t(q_i)$  are the partial derivatives of image  $I$  with respect to  $x$ ,  $y$ , and  $t$  (time) at the pixel instance  $q_i$  and  $V_x$  and  $V_y$  are the image flow (velocity) vectors (Equation 3.9). This equation is for a single image  $I$  and will be applied across the pixels from  $q_1$  to  $q_n$ .

$$I_x(q_i)V_x + I_y(q_i)V_y = -I_t(q_i) \quad (3.9)$$

With a system of  $n$  equations and two unknowns, this system is overdetermined and requires another step for determination of the system. Applying the least squares principle, the Lucas-Kanade tracking equation solves for the velocity vector with the sums iterated from  $i$  to  $n$  in the image  $I$  (Equation 3.10).

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} \sum_i I_x(q_i)^2 & \sum_i I_x(q_i)I_y(q_i) \\ \sum_i I_x(q_i)I_y(q_i) & \sum_i I_y(q_i)^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_i I_x(q_i)I_t(q_i) \\ -\sum_i I_y(q_i)I_t(q_i) \end{bmatrix} \quad (3.10)$$

The velocity vectors computed contain the iterative movement of the selected points through the image sequence. The OpenCV function used for this was *calcOpticalFlowPyrLK*. This sparse iterative method uses pyramids to reach convergence among the residual differences between the images (Bouquet 2001). The parameters are: Shi-Tomasi feature points, window size, maximum level, and termination criteria. Window size is the size of the search window used to search for the feature points across the images, which can be large or small based on the expected amount of movement of the features. Maximum level sets an upward limit on the number of pyramid levels to be used. Termination criteria defines the termination point of the iterative search algorithm, which is based on when it reaches a maximum number of iterations or when the search window moves by less than a minimum distance  $\varepsilon$  in pixels.

The tracked velocity vector of the features can then be corresponded to a displacement. If the original location of the pixel in  $(x, y)$  cartesian coordinates is  $(a, b)$  and the point moves to position  $(c, d)$ , as noted by the velocity vector, the motion of the point is the difference between the original and new points, per Equation 3.11 and as illustrated in Figure 3.10. The motion of the object is quantified by the change in  $x$  ( $dx$ ) and change in  $y$  ( $dy$ ). For a series of feature points on a rigid body, the average of all tracked points can be taken to estimate an overall motion in  $x$  and  $y$  for the rigid body. The illustration in Figure 3.10 tracks the motion of a single point of an object over time where each point of motion of the object represents a new image. This leads to results of  $x$  and  $y$  movement between each pair in a set of images. The initial results of this are found in

pixel coordinates, which can then be scaled to physical units. Displacement scaling occurs by applying a scale factor based on a known distance in the images.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a - c \\ b - d \end{bmatrix} \tag{3.11}$$

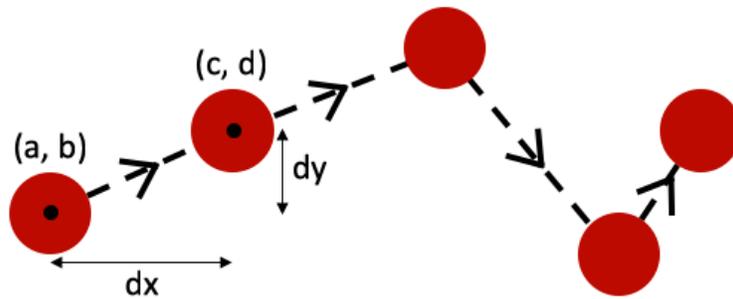


Figure 3.10. Tracking of an object through space using optical flow.

Overall, this process of optical flow estimation using multiple points on an object over time can be visualized through Figure 3.11. This graphic illustrates the process of a template image being taken, the selection of high-contrast corner feature points, and the iterative tracking of those feature points in  $x$  and  $y$  through  $n$  subsequent images.

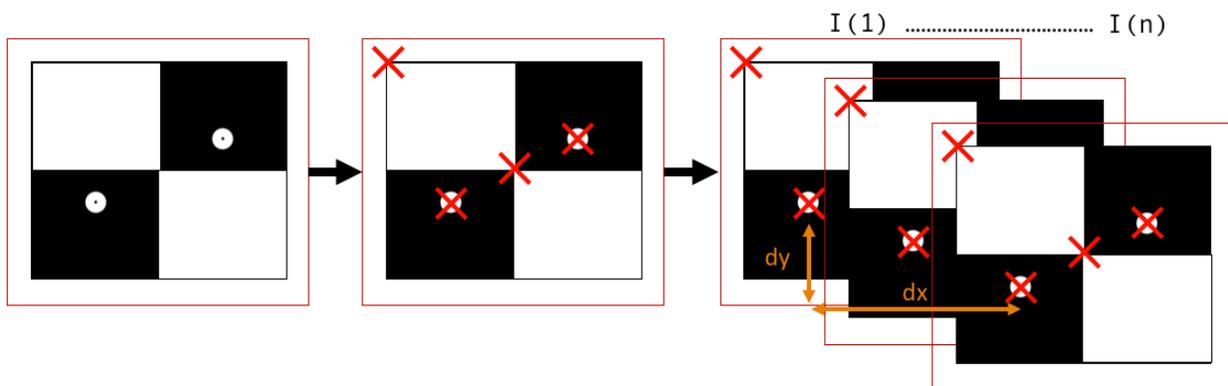


Figure 3.11. Feature tracking across a set of images using optical flow.

The use of the Lucas-Kanade-based optical flow method relies on three main assumptions. These assumptions are: (1) movement of the object is small and predictable between images (constant flow); (2) all motion is in-plane and 2D; and (3) brightness is consistent between images (Kim et al. 2009; Lim et al. 2005). These assumptions limit the robustness of the tracking, but do

not limit the applicability of this method to the tracking of displacements in civil infrastructure. Slow, predictable movement typically fits within the realm of testing of civil infrastructure. Though movement is never truly predictable, it can be expected to a certain degree (e.g. a bridge girder will deflect vertically under the load of a truck). Most rigid body targets (e.g. bridge girder, traffic signal structure, etc.) will act this way. A strong image acquisition rate supports the ability of the images to capture all movement of the object and avoid any image aliasing. Temporal aliasing occurs when the frequency signal of the target object is greater than the camera frame rate, which results in gaps in the motion tracking of the object (Feng and Feng 2018). To mitigate this, the camera acquisition rate should be at least twice the expected frequency of the target per the Nyquist frequency criterion (Bartilson et al. 2015). For rigid structures (such as a bridge girder), natural frequencies are low and no significant local 3D or out of plane movement is expected. For full field measurement of a structure, where measurement of local or global 3D motions is desired, other measurement systems should be used. Finally, brightness can be maintained between images by manipulating the ambient lighting with lights and using a high contrast target that is easily seen against its background even with ambient changes in lighting. In a lab setting, lighting can be fully controlled. Outdoor field tests should be conducted during short periods of consistent natural light to facilitate the greatest brightness consistency between images.

The benefits of the optical flow method outweigh its limitations. Unlike correlation methods, optical flow-based tracking does not require the use of a patterned target card. This creates the opportunity for tracking of high-contrast natural features. Additionally, this algorithm is computationally fast and, because neither the feature points nor the images themselves are dependent on each other, opportunities for parallel computing arise (Kim et al. 2009). A natural tradeoff between robustness and accuracy exists; however, previous research studies using optical flow-based methods have proven to be relatively accurate. Two field tests resulted in error resolutions of 0.1 to 0.4 mm for cable vibration measurement (Ji and Chang 2008) and 2.14 mm for bridge displacement measurement using an unmanned aerial system (Yoon et al. 2018).

## **4 Experimental Validation**

### **4.1 Motivation**

The previous chapter proposed a vision-based displacement measurement system using feature tracking. Next, this proposed system was validated through a series of laboratory studies and field testing. The purpose of this validation was to ascertain the conditions necessary for applying the system, the overall accuracy and reliability of the methods, and the limitations inherent to the key assumptions. Section 4.2 will present the final design of the system based on the methods introduced in Chapter 3. Section 4.3 will introduce a series of four lab tests to evaluate both the hardware and software aspects of the system among the unique factors that are present in vision-based measurement systems. Section 4.4 will present a field test that applies this system to contactless displacement measurement of a civil structure. The results from these lab and field studies will be presented in the following chapter.

### **4.2 Final System Design**

This vision-based system entails image acquisition and image processing for displacement measurement. Among the methods presented in Chapter 3, the final system features the NVIDIA Jetson TX2 as the main computing device, the FLIR Grasshopper3 (GS3) camera and a Tamron lens (various models) for image acquisition, and the Lucas-Kanade-based optical flow method (OFM) for image processing in order to produce measurement results of pixel displacement. Using a scale factor, this pixel displacement can be scaled to physical units. Validation of the system will include comparison of the OFM displacement results to that of a normalized cross-correlation algorithm as well as a traditional displacement measurement device. Figure 4.1 illustrates this process.

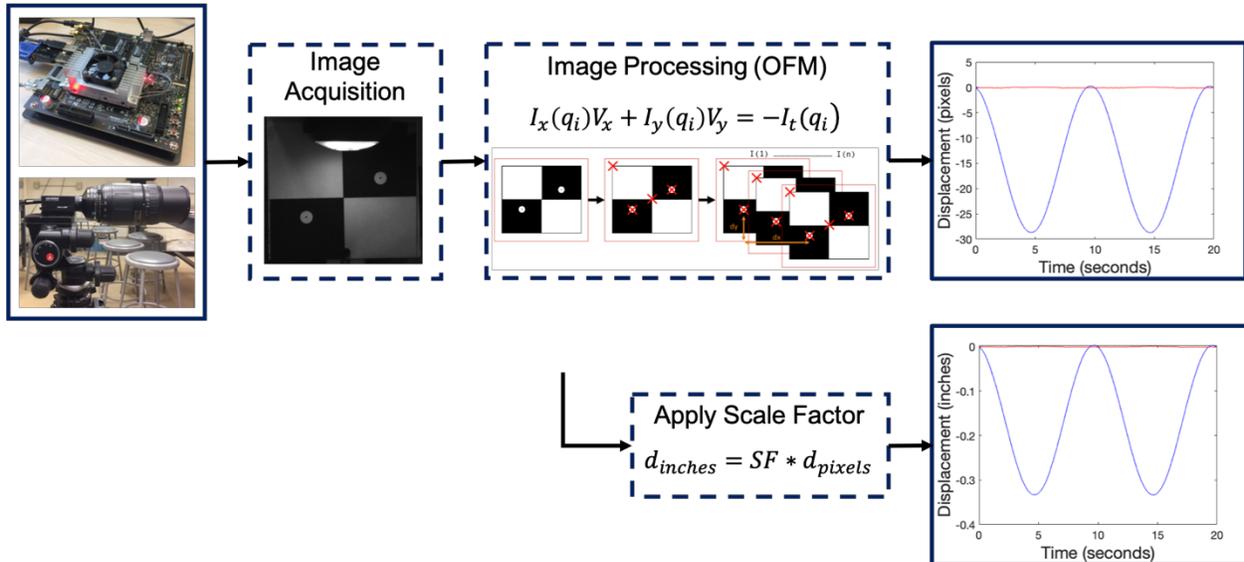


Figure 4.1. System process diagram.

### 4.3 Lab Testing

After the system was designed, experimental testing was conducted to contribute to the overall development of the system by testing the components together and gaining an understanding of the parameters required. A series of lab tests were performed to evaluate the feasibility and reliability of the proposed system under various controlled conditions. First, the initial feasibility of the system was studied. Subsequent testing focused on lighting conditions, tracking of natural features, and comparison against a ground truth displacement measurement. These tests will be introduced in detail in the following sections.

#### 4.3.1 Feasibility Study

For this feasibility study, the purpose was to collect a standard set of images using the GS3 camera controlled by the Jetson TX2. Because of this, the variables studied were limited to frequency of the target, lighting, and the target pattern. Additionally, previous work had proven the effects and applicable correction methods for factors such as camera tilt, eccentricity, blurriness, and speckle pattern size under a similar system configuration (Alipour et al. 2019).

Two black and white target cards were created for the experimental tests based on designs from similar vision-based testing, some of which were presented in Section 2.3.1. As mentioned in Section 3.2.2, the GS3 camera at full focal length (300 mm) and 30 ft was capable of a minimum FOV of 13.6 x 8.5 in. As this did not fit within the 6.5 x 6.5 in. FOV of previous testing, a 14 in.

window was used for the image widths, but the target size of 6 x 6 in. was still used. Essentially, this resulted in the images viewing more than just the target, which opened up the potential for background noise. To avoid the consequences of this, test backgrounds must be well selected or images can be cropped as needed during post-processing.

To optimize area-based tracking, one target was a random speckle pattern. The previous study used a speckle size of 2 mm, which came from a range of desired speckle size of 10-15 pixels. Applying the equations from Section 3.3.2 to this application, an image size of 1920 x 1200 pixels from the GS3 (2.3 MP) at the field of view of 14 in. and desired pixel size of 10-15 pixels determined a speckle size between 1.85 and 2.78 mm. Based on this range, the speckles were sized to a median value of 2.40 mm. This was within the range of acceptable speckle size (greater than 0.5 mm and less than 3 mm) determined by the specific parameters of the previous study (Alipour et al. 2019). Specifying a desired speckle size of 10-15 pixels ensured that the object (speckles) were large enough for accuracy of pixel level tracking by creating sufficient regions of unique and distinct patterns. Target 1, shown in Figure 4.2, features the random speckle pattern of 2.40 mm speckles that was generated digitally.

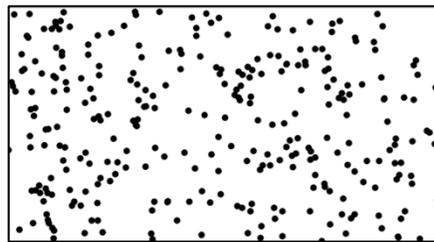


Figure 4.2. Speckle pattern.

To optimize feature-based tracking, the second target was a checkerboard pattern with distinct points of contrast. Target 2, shown in Figure 4.3, featured a checkerboard with two white circles and a single black dot within them. This pattern was designed to create several high-contrast points of interest that could serve as trackable features. These main points of interest were the center intersection of the checkerboard and the black dots centered in the two white circles. The black dots are 1.70 mm and the diameter of the white circles are 13 mm, which were sized visually to meet a desirable appearance. Both targets were printed and placed on metal cards that were 6 x 6 in.

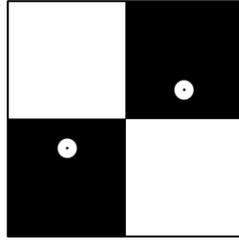


Figure 4.3. Checkerboard pattern.

A motorized linear translation stage manufactured by Zaber Technologies Inc. was used to control the movement of the targets. The target cards were supported on the translation stage by a metal bar that was fixed to the stage. Photos of the target cards mounted onto the translation stage are shown in Figure 4.4. The specifications for the translation stage are documented in Table 4.1. The motion of the translation stage was programmed so that it did not need to be managed manually.

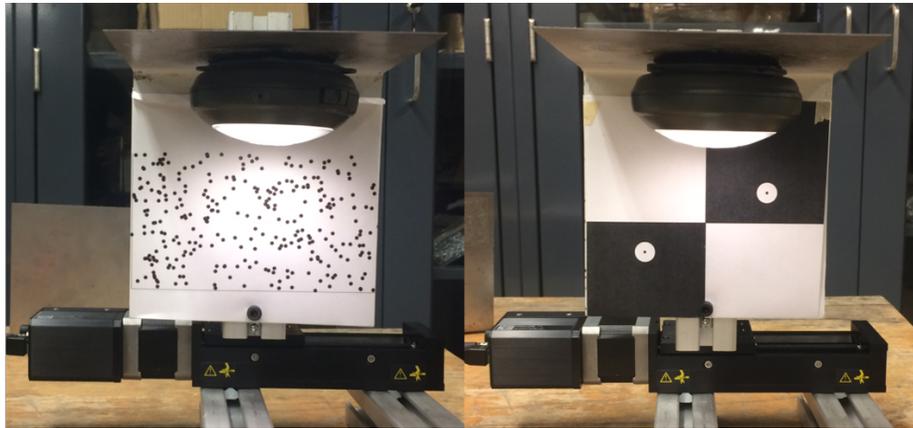


Figure 4.4. Target cards mounted on translation stage.

Table 4.1. Zaber translation stage specifications.

<b>Component</b>	<b>Specification</b>
Model	Zaber X-LSQ075-E
Maximum Distance	75 mm
Maximum Speed	280 mm/s
Accuracy	45 $\mu$ m

As pictured, an LED light was attached to the cards to indirectly light the targets to create an optimal level of brightness. The camera was always properly focused and perpendicular to the face of the target card. The eccentricity of the setup was minimal and a visual of the distance (30

ft) between the camera and the target is shown in Figure 4.5. The testing matrix featuring all 12 tests with the varied parameters is shown in Table 4.2. The translation stage was programmed to move the target an amplitude of 0.5 in. every test. The speed of the targets was varied between 0.1 Hz (slow), 1.0 Hz (medium), and 3.0 Hz (fast). Half of the tests were conducted with the room lights off and the other half were conducted with the presence of ambient room lighting (both with the LED). The same scenarios were tested for each target pattern. The lens used was the Tamron A17 with an aperture setting of f/8 for all tests. The rate of image acquisition was 10 fps. Overall, the factors studied are summarized as system connectivity (ability of Jetson to control image acquisition), speed of capture related to speed of target, lighting, and target pattern.

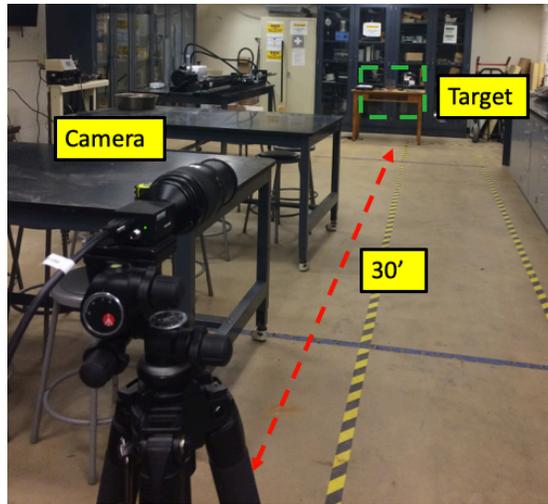


Figure 4.5. Feasibility test setup.

Table 4.2. Feasibility test matrix.

Test No.	Frequency (Hz)	Lighting	Target Card
1	0.1	Dark	Speckle
2	0.1	Ambient	Speckle
3	1.0	Dark	Speckle
4	1.0	Ambient	Speckle
5	3.0	Dark	Speckle
6	3.0	Ambient	Speckle
7	0.1	Dark	Checkerboard
8	0.1	Ambient	Checkerboard
9	1.0	Dark	Checkerboard
10	1.0	Ambient	Checkerboard
11	3.0	Dark	Checkerboard
12	3.0	Ambient	Checkerboard

#### 4.3.2 Lighting Study

Lighting is one of the most critical factors that affects the efficacy of vision-based measurement systems. In the initial feasibility tests in Section 4.3.1, lighting was varied by taking images under ambient room lighting and in a dark room, both with an LED light to illuminate the target card. This proved to affect the quality of the images and therefore the results of the image processing. This variability of lighting conditions furthered interest in assessing the effect of the camera settings on the quality of images and the subsequent ability of the OFM algorithm to process them accurately. This new series of tests would serve to determine the ideal image quality settings for this controlled application of the OFM algorithm.

The main lighting parameter from the camera that can be manipulated is the aperture setting. Aperture is a hole that controls the amount of light allowed into the camera sensor. The size of the aperture is one of the factors that affects depth of field in images. The ideal aperture setting for a given scenario may vary based on the external lighting, exposure, and the desired visibility of the images. Some cameras have a fixed aperture setting, while others allow for manual modification of the aperture setting. As introduced in Section 3.2.2, the Tamron 672D lens has seven exposure settings:  $f/4$  (wide aperture, brightest exposure),  $f/5.6$ ,  $f/8$ ,  $f/11$ ,  $f/16$ ,  $f/22$ , and  $f/32$  (small aperture, darkest exposure). Thus, testing was completed with the Tamron 672D to evaluate the effect of change in aperture setting on the images. For this test, aperture settings in the middle range were studied:  $f/8$  and  $f/16$ . The settings  $f/4$ ,  $f/5.6$ ,  $f/22$ , and  $f/32$  were deemed extreme and

not necessary to be iterated on as part of testing; however, sample images at each of these levels were taken to evaluate their effect on visibility of the images.

This testing repeated the setup of the feasibility tests conducted in Section 4.3.1 using the speckle and checkerboard pattern targets under dark and ambient lighting conditions, as listed in Table 4.3. The distance between the camera and the target was 30 ft. The targets moved on the translation stage at “slow” and “medium” speeds of 0.1 Hz and 1.0 Hz, respectively, for the extent of the translation stage distance. All of the feasibility tests in Section 4.3.1 were conducted with the aperture setting of f/8, so all new images here were taken at f/16 for comparison of the two sets of images. The faster tests (3.0 Hz) were not repeated. All images were captured at the previous rate of 10 fps.

Table 4.3. Lighting test matrix.

<b>Test No.</b>	<b>Frequency (Hz)</b>	<b>Lighting</b>	<b>Target Card</b>
1	0.1	Dark	Speckle
2	0.1	Ambient	Speckle
3	1.0	Dark	Speckle
4	1.0	Ambient	Speckle
5	0.1	Dark	Checkerboard
6	0.1	Ambient	Checkerboard
7	1.0	Dark	Checkerboard
8	1.0	Ambient	Checkerboard

#### 4.3.3 *Natural Feature Tracking Study*

The next series of lab tests were conducted to evaluate the ability of the algorithm to track natural features instead of a patterned target card. This enables contactless measurement if high-contrast features that are already part of the target structure can be tracked, which removes some of the field test difficulties imposed by the previous digital image correlation-based method. Such trackable features include concrete cracks, steel bolts, and high-contrast edges of a target object.

For this testing, a T-shaped steel section and cracked concrete cylinder were used as targets, as shown in Figure 4.6. The steel section was evaluated from both a forward facing and a side orientation. From the forward view, the steel section was 104.5 mm wide. From the side view, the steel section was 90.0 mm wide. The concrete cylinder had a diameter of 102.3 mm in the visible orientation. The setup followed that of previous tests with the targets at 30 ft away from the camera. The amplitude of the motion of the translation stage was 0.5 in. at speeds of 0.1 Hz and 1.0 Hz.

The scenarios tested are shown in Table 4.4. The Tamron 672D lens was used and the aperture setting was varied between f/8 and f/16 for continued evaluation of the optimal lighting conditions on these new targets. The LED light was again used to indirectly light the targets. Images were acquired using the GS3 at 10 fps.

This stage of testing was designed for the purpose of tracking of the bolts of the steel section and the crack on the concrete cylinder. A solid background was placed during this test to remove any discrepancies due to points of contrast in the background of images.

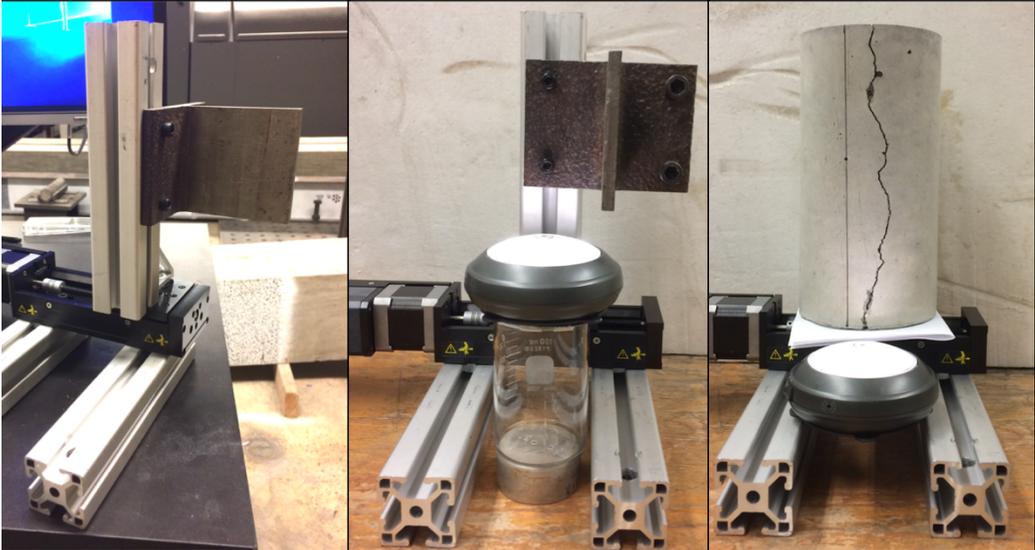


Figure 4.6. Steel section (side and forward orientations) and concrete cylinder on translation stage.

Table 4.4. Natural feature test matrix.

Test No.	Frequency (Hz)	Lighting	Aperture	Target
1	0.1	Ambient	f/8	Steel (side)
2	1.0	Dark	f/8	Steel (side)
3	1.0	Ambient	f/8	Steel (side)
4	1.0	Ambient	f/16	Steel (side)
5	0.1	Dark	f/8	Steel (forward)
6	0.1	Ambient	f/8	Steel (forward)
7	1.0	Dark	f/8	Steel (forward)
8	1.0	Dark	f/16	Steel (forward)
9	1.0	Ambient	f/8	Steel (forward)
10	1.0	Ambient	f/16	Steel (forward)
11	0.1	Dark	f/8	Concrete
12	0.1	Dark	f/16	Concrete
13	0.1	Ambient	f/8	Concrete
14	0.1	Ambient	f/16	Concrete
15	1.0	Dark	f/8	Concrete
16	1.0	Dark	f/16	Concrete
17	1.0	Ambient	f/8	Concrete
18	1.0	Ambient	f/16	Concrete

#### 4.3.4 Comparison Against Traditional Displacement Measurement

Lab testing up to this point focused on determination of the ideal image acquisition settings for OFM processing with comparison of results to the translation stage’s ideal sinusoidal motion and a conservative cross-correlation algorithm. In addition, a comparison with a traditional displacement measurement system was desired to serve as a ground truth validation for the proposed vision-based displacement system. To conduct this validation, the next phase of testing used a string potentiometer to obtain comparative displacement measurements. A string potentiometer (“string pot”) is a traditional displacement measurement system which features a cable that extends to measure linear position. The string pot used in this study was manufactured by Bridge Diagnostics, Inc. (BDI) and has been used in previous bridge load testing field studies, as shown in Figure 4.7. When mounted to the bottom of a bridge girder, it can measure the vertical deflections of the girder. However, it requires a fixed ground point, such as a camera tripod, to support it. Depending on the conditions underneath a bridge, this can pose difficulties during instrumentation that limit the ability to use such a device. Additionally, the displacement wire must be calibrated so that it is taut and does not sag, which would lead to imprecise measurements.

These required conditions already prove some of the advantages of a vision-based system that does not require attachment to the structure and that can measure displacements in two directions.

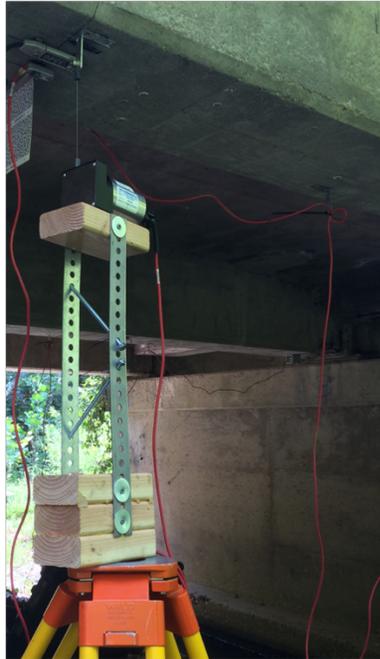


Figure 4.7. Extended string potentiometer supported by tripod under bridge girder.

Instead of using the programmed translation stage, a cantilever frame was built in order to have a structure that would support semi-controlled motions and attachment of a string pot. The frame primarily allowed horizontal motions, so the string pot was attached horizontally for this measurement. Whether the string pot collected either vertical or horizontal motions was not deemed to be a significant factor in the testing, as long as it could measure one of the two. Manual excitation of the cantilever in primarily the horizontal direction, with some inherent vertical motion due to the cantilever not being truly fixed longitudinally at its support, created non-linear horizontal displacements. The length of the cantilever increased the amount that the frame moved. A smaller cantilever was considered, but might have been too stiff within the frame connection to create enough quantifiable movement. Some inherent difficulty existed with the string pot because the tension in the cable sought to dampen the cantilever frame motion, but upon evaluation of this it was decided that extended periods of motion were unnecessary, so this factor did not affect the suitability of the setup.

The GS3 camera with Tamron 672D lens was set to acquire images at 10 fps. The BDI string pot data acquisition rate was 100 Hz, with measurements of displacement in inches. The distance between the camera and the target was 30 ft, which followed the setup of previous tests. The overall test setup is shown in Figure 4.8. The parameters varied during the tests are shown in Table 4.5. The cantilever was manually excited by an operator shaking it at two motion amplitudes: “large” and “small.” Most tests were conducted for approximately 5 seconds, while two lasted for up to 20 seconds. The lighting and aperture settings varied as usual, with ambient and dark room lighting, both with an LED to illuminate the target, though no new results were expected from these. The checkerboard pattern was connected to the end of the cantilever as the primary target. The string pot cable was connected to the end of the cantilever on the back face of the target card, making them essentially on the same plane.

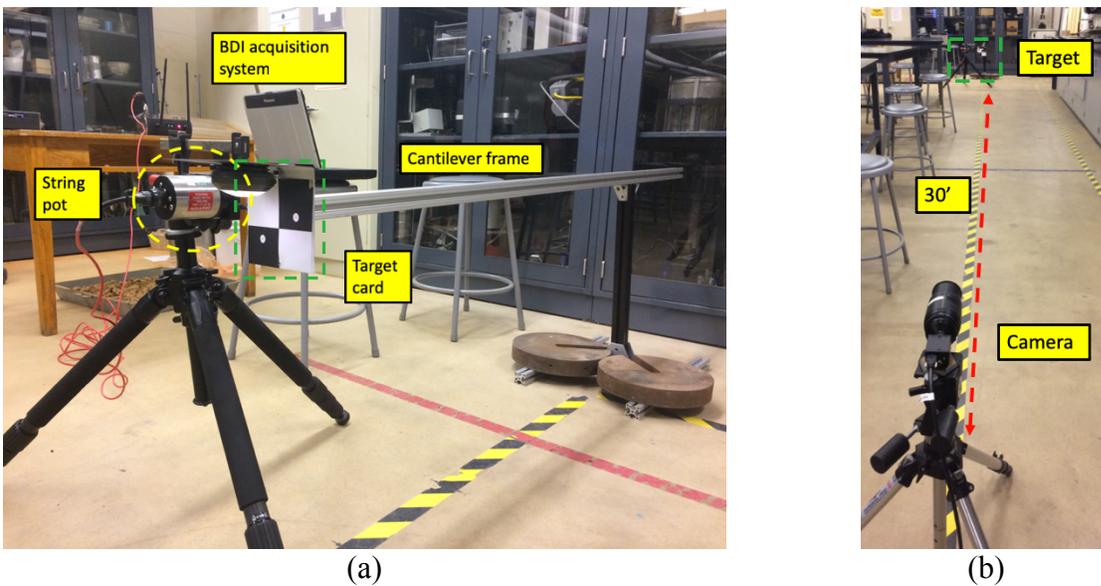


Figure 4.8. String pot comparison test setup: (a) Close-up of target and (b) Camera to target view.

Table 4.5. String pot comparison test matrix.

Test No.	Excitation	Lighting	Aperture	Target
1	Large	Dark	f/8	Checkerboard
2	Large	Dark	f/16	Checkerboard
3	Large	Ambient	f/8	Checkerboard
4	Large	Ambient	f/16	Checkerboard
5-6	Small	Dark	f/8	Checkerboard
7	Small	Dark	f/16	Checkerboard
8-9	Small	Ambient	f/8	Checkerboard
10	Small	Ambient	f/16	Checkerboard

#### 4.4 Field Testing

A field test was conducted to validate the proposed system in an uncontrolled environment. The progress from lab testing influenced the factors studied through the field test. The field test entailed a full implementation of the system for measuring displacement of a traffic signal structure.

##### 4.4.1 Traffic Signal Structure Deflection Study

Traffic signal structures pose difficulties for instrumentation of traditional sensors. Locations of interest are typically tall and over active roadways, so instrumenting them requires stopping traffic, use of a crane for access, use of fall protection, etc. Direct application of displacement sensors, such as string pots that require fixed ground, is especially difficult under these circumstances, so accelerometers would likely be used. From the vibration data and known modal parameters, vertical displacement can be estimated, though this has not always proven to be a reliable method for obtaining displacements (Brown et al. 2019). Amidst these difficulties, reasons exist to study the motions of traffic signal structures. If such motions are above a certain threshold, typically due to environmental loads, the owner may be interested in taking preventative measures to stabilize the flexible structure to avoid buildup of fatigue due to this cyclic stress. Thus, the application of vision-based measurement technology creates new opportunities for monitoring such structures (Bartilson et al. 2015).

Measurement of the end displacement of a traffic signal structure mast arm evaluated the ability of this proposed vision-based system to acquire measurements in an uncontrolled field setting. The system was applied to track ambient wind excitations without a reference of traditional sensors. The traffic signal structure chosen for monitoring rests above the westbound lanes of

Fontaine Ave at the intersection of Fontaine Ave and Ray C Hunt Dr in Charlottesville, Virginia. A plan view of this intersection, with the instrumentation location and the structure under consideration noted, is shown in Figure 4.9.

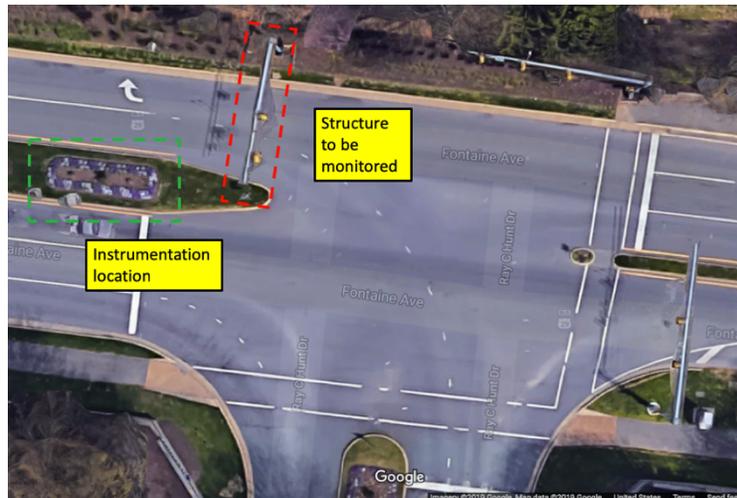


Figure 4.9. Plan view of Fontaine Ave at Ray C Hunt Dr (adapted from Google Maps 2019).

This contactless testing utilized the FLIR GS3 camera and Tamron 672D lens controlled by the NVIDIA Jetson TX2. The system was placed in a median separating the east and westbound lanes of the roadway at a distance about 45 ft away from the traffic signal structure, which was at a height of 25 ft. Additionally, the GoPro Hero 3+ with the Tamron A17 lens was utilized as a second acquisition source. The cameras were at a height of 46.5 in. (3.875 ft) and aimed at the end of the mast arm. This created a camera-to-target angle of 25.15 degrees from the horizontal and a perpendicular distance of 49.71 ft between the camera and the target. Instead of attaching a target card to the structure, the end of the mast arm was to be used as a region with trackable feature points for this contactless testing. Over one foot from the end of the mast arm, there was a 12 x 18 in. traffic sign that consisted of many trackable features. If displacement was obtained at the location of the sign, the displacement at the very end of the mast arm could be extrapolated based on assumption of linear increase in displacement along the length of the cantilever. Figure 4.10 and Figure 4.11 illustrate the instrumentation setup and provide annotations of the dimensions. Applying the field of view equations from Section 3.2.2 to this distance (49.71 ft) and with the full focal length of the lenses (300 mm), the field of view of the GS3 camera was 22.5 x 14.1 in. and that of the GoPro camera was 8.0 x 10.7 in. As these dimensions represent the minimum possible

camera view, they show that both cameras were appropriate for obtaining the desired field of view of a sign that is 18 in. With a nominal margin around the sign, the images will view the sign within the limits of the maximum focal length.



Figure 4.10. Traffic signal structure test instrumentation setup.

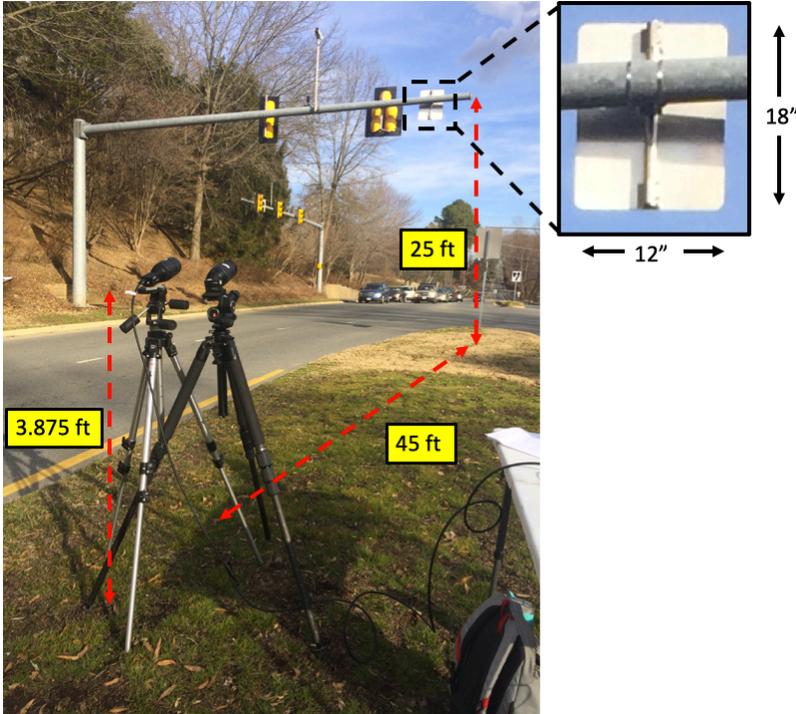


Figure 4.11. Traffic signal structure test setup with dimensions.

The GS3 was set to an image acquisition rate of 10 fps. As noted in Section 3.2.2, the GoPro is capable of both image and video acquisition. For this purpose of secondary acquisition, the GoPro was set to record color video at a rate of 60 fps. As video frames can be extracted for image processing, this would enable another reference for tracking of image frames in case the 10 fps rate was within the frequency of motion of the structure and led to image aliasing. The GoPro app running on an iPad controlled the video acquisition through a visual interface. Aperture settings of f/8 and f/16 were utilized on the Tamron 672D-GS3 and the fixed setting of f/8 was utilized on the Tamron A17-GoPro.

Images and video were collected during six periods of environmental loading, as listed in Table 4.6. Half of the tests were conducted for a period under 20 seconds and the others were less than 60 seconds, which was chosen based on the amount of motion exhibited by the structure. During the testing period, wind excitation was low, with an average wind speed of 1 mph SSW and with gusts up to 5 mph per the daily wind data for Charlottesville-Albemarle (WindAlert 2019). However, it was observed that the movement of large passenger vehicles and box trucks under the structure excited the structure more than the wind alone. Thus, acquisition periods were chosen to align with such vehicles approaching the intersection, which led to the most visible vibration of the structure. The late afternoon setting optimized the ambient lighting, which was not too bright or too dark (fairly consistent). Table 4.6 also documents the variation of the aperture of the GS3, which was calibrated based on the ambient lighting brightness (f/16 ultimately chosen to create darker images given the amount of ambient light).

Table 4.6. Traffic signal structure test camera acquisition periods.

<b>Test No.</b>	<b>Acquisition Time (seconds)</b>	<b>Tamron 672D Aperture Setting</b>
1	17	f/8
2	53	f/16
3	17	f/16
4	17	f/16
5	52	f/16
6	52	f/16

## 5 Experimental Results

The previous chapter introduced four controlled lab studies and one field test. In this chapter, the results from these tests will be presented and discussed. Section 5.1 will present the lab test results and Section 5.2 will present the field test results. Evaluating the performance of the proposed system under a variety of test conditions will help to draw conclusions about its feasibility for further applications and open up the discussion for recommended next steps, which will be presented in the following chapter.

### 5.1 Lab Testing Results

Results from all of the lab tests described in Section 4.3 will be presented in the following sections. The results from the lab tests will determine the optimal conditions for the use of the vision-based system in order to prepare the system for field testing applications. Results presented will primarily take the form of visual inspection of the quality of the optical flow tracking, correlation of displacement between image processing methods, or comparison of displacement with a reference instrument.

#### 5.1.1 Feasibility Study Results

The purpose of the feasibility study was to calibrate the image processing parameters (inputs for feature selection and tracking) and the test conditions (image acquisition rate, target speed, lighting, target pattern). The results for this calibration are presented through visual inspection of the optical flow tracking and correlation of displacement results between the two image processing algorithms presented previously.

The baseline set of images were processed using the Lucas-Kanade-based optical flow method (OFM) per the functions introduced in Section 3.3.3. As a sensitivity study for the algorithm parameters, all 12 sets of test images were processed under five iterations. The parameters studied for both feature selection and tracking for each of the iterations are listed in Table 5.1. The parameters listed here were defined in Section 3.3.3. For the feature selection parameters, maximum corners was maintained as a high upper limit to encourage the return of all strong features and not reject the selection of any good features. Quality level, minimum distance, and block size were varied. As these three parameters increased, the number of points selected decreased. For the feature tracking parameters, window size was varied and maximum level and

termination criteria remained constant. Increasing window size improved the ability of the tracking to follow the movement of features, with a tradeoff of speed.

Table 5.1. Iteration of Lucas-Kanade-based optical flow method processing parameters.

Iteration	Feature Selection				Feature Tracking		
	Max Corners	Quality Level	Min Distance	Block Size	Window Size	Max Level	Termination $\epsilon$   Count
1	100	0.3	7	7	15, 15	2	0.03   10
2	100	<b>0.9</b>	<b>20</b>	<b>20</b>	15, 15	2	0.03   10
3	100	0.9	<b>12</b>	<b>12</b>	15, 15	2	0.03   10
4	100	<b>0.5</b>	12	12	15, 15	2	0.03   10
5	100	0.5	12	12	<b>30, 30</b>	2	0.03   10

The first iteration began with a sample set of parameters. This consisted of a very low quality level and small required distance between features, so many features were selected. The results outputted were the averages of the horizontal and vertical motion of all of the feature points. In the second iteration, the parameters were refined to be more selective so that fewer points would be tracked. This involved increasing the quality level, minimum distance between points, and block size. The number of features tracked became very sparse, so these parameters were then decreased below this threshold. In the third iteration, the quality level was maintained and the area selection parameters were decreased to 12 to allow for less space between feature points. This led to a small number of features being selected, but the ones that were chosen were strong. Thus, in the fourth iteration, the quality level was lowered to create a less conservative selection of features. This would serve to increase the number of features selected, which would create a larger sample size for more accurate measurement when points were all averaged together. These first four iterations were focused on the sensitivity of the feature selection parameters. After this calibration led to an agreeable selection of features, the final iteration evaluated change in a feature tracking parameter. The tracking window size was increased to improve the consistency of tracking of the feature points based on their expected movement. This increased the flexibility of the tracking based on the amount of movement expected between images, at a minor loss of speed due to the need to scan a larger region. This final iteration saw a convergence of the results, so the parameters used in it were applied to all subsequent image processing.

The parameter sensitivity study sought to optimize the tracking results for all images; however, it was discovered that the checkerboard was more successfully tracked than the speckle

pattern. An example of the optical flow tracking of both targets under the final parameters is illustrated in Figure 5.1. The results revealed that tracking performance for the speckle pattern was consistently poor due to the point-based nature of the optical flow algorithm. OFM looks for high contrast points and because the speckle pattern consists of black dots on a white background, there is an excessive number of high contrast points. This does not limit the ability of the feature selection, but ensures that the feature points that are selected are less unique. This means that the feature point can be hard to follow between frames when there are multiple points within its tracking window that look exactly the same. The points may jump around, essentially tracking the displacement of a combination of locations in one velocity vector. This leads to extremely inconsistent tracking that is mostly noise. Conversely, the checkerboard design allowed for the algorithm to follow the centers of the circles and the center of the card because these features were distinct and located outside of the minimum distance from each other.

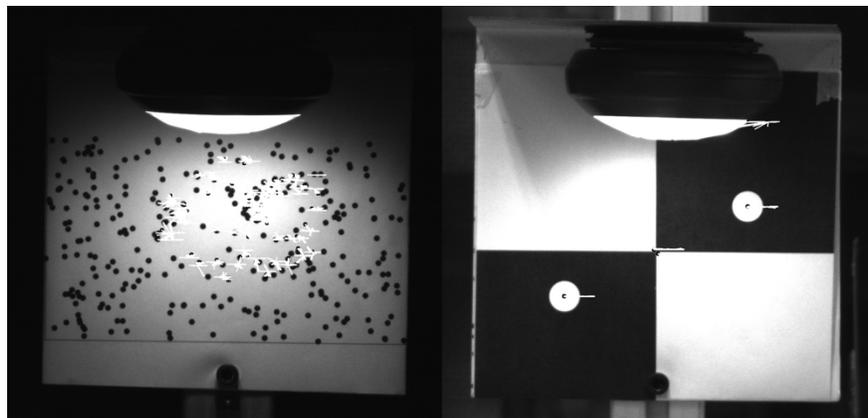


Figure 5.1. Feasibility test: Sample optical flow tracking for speckle and checkerboard target patterns.

In the sample tracking shown in Figure 5.1, the feature points tracked on the speckle pattern included a collection of points on the white background and adjacent to the speckles, as well as some speckles themselves. The results of the feature point selection here are based on the level of contrast identified between the speckle points and the surrounding white region. The tracking of these points showed some linearity, but also saw vertical jumps when the tracked points were misidentified. The feature points tracked on the checkerboard pattern included the centers of both circles, two points at the intersection of the center of the target, and one point on the LED light. The tracking of these first four points was primarily linear, aligning with the motion of the target.

The tracked point on the LED light showed some erratic motion and the reasons for this will be discussed later. As noted, a smaller number of distinct points proved more consistent than a larger number of similar features grouped closely together, as occurs in the speckle pattern. Thus, the final parameters selected favored tracking of the checkerboard pattern.

To provide a comparison for OFM as an image processing method, a normalized cross correlation (NCC) algorithm was applied to the image sets using the criteria presented in Section 3.3.2. NCC has similarities to previous applications of digital image correlation (DIC) (Alipour et al. 2019) and its comparison with OFM can be justified because it is a conservative area-based algorithm. Where OFM is weak, NCC is strong, and this dichotomy will serve to validate both image quality and OFM results. For example, if NCC produces realistic displacement results and OFM is poor, it is likely that the images are good and there is a weakness with OFM. If NCC and OFM are both off, the images being processed may be of poor quality. If NCC and OFM both produce visually accurate results, it confirms both the images and the algorithms.

A generous region of interest (ROI) was manually selected for each set of images to ensure that the tracked points would remain in the frame given the extent of the target movement. The size of the ROI, shown in Figure 5.2 for both targets, decreased the speed of the tracking, but increased the accuracy of the results. As an example of the processing speeds occurring in this study, the OFM processing occurred at an average rate of 22.2 images/sec for the speckle pattern and 24.4 images/sec for the checkerboard. The NCC processing occurred at a significantly slower rate of 0.70 images/sec for the speckle pattern and 0.66 images/sec for the checkerboard. However, this comparison of processing speeds has little merit for this study as the NCC method was not optimized for speed but for accuracy by processing a large region of the image. The OFM processing, contrarily, had not yet been optimized for accuracy. Thus, a more comprehensive comparison of processing speeds between the methods will be presented in a following section.

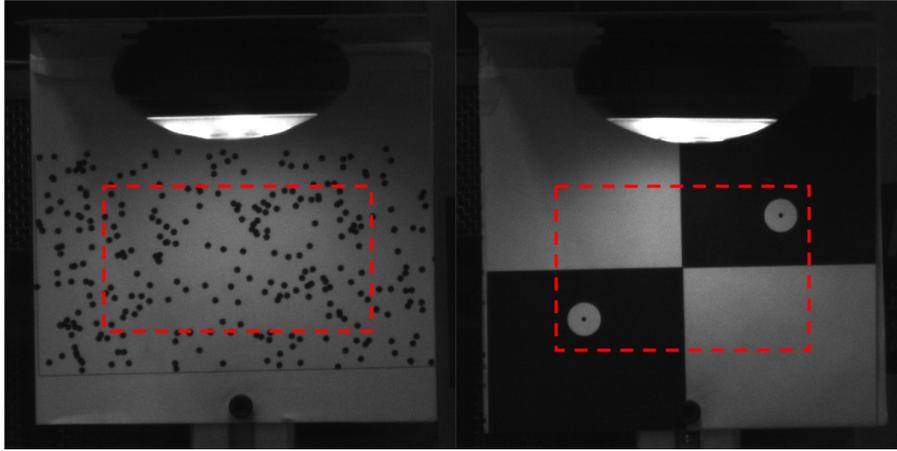


Figure 5.2. Original ROI for speckle and checkerboard target patterns.

Sample displacement plots comparing the OFM and NCC results for the tests from which the tracking results presented in Figure 5.1 originate are shown in Figure 5.3 and Figure 5.4. The plots include the sinusoidal shape of the 1.0 Hz idealized motion of the translation stage for reference. All displacements were scaled from pixels to physical units in inches using the dimensions of the target card (6 in. horizontal). A small amount of error can be produced from this determination of the scale factor, but it would be on the order of pixels in this study and is considered negligible.

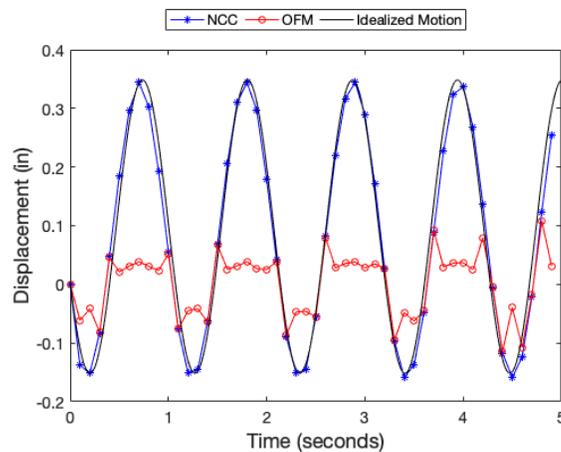


Figure 5.3. Feasibility test: Sample horizontal displacement for speckle target pattern.

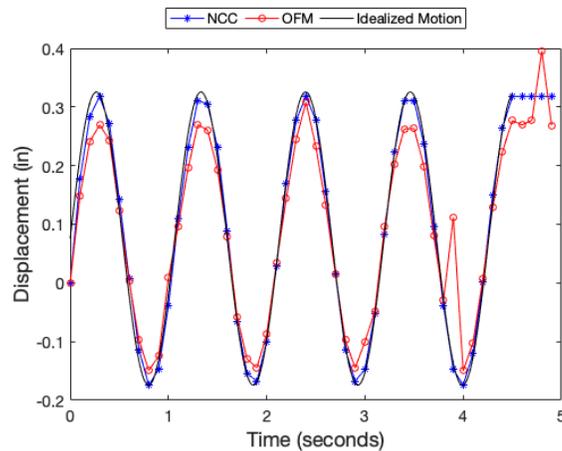


Figure 5.4. Feasibility test: Sample horizontal displacement for checkerboard target pattern.

The results show that the NCC processing was able to track the motion better for both targets when compared to OFM. The NCC algorithm successfully tracked movement in all of the images and was unaffected by variations in lighting. OFM tracking was lacking due to factors in both image acquisition and image processing. The OFM tracking result was particularly poor for the speckle pattern tests and the reasons for this were discussed previously. As an area-based technique, NCC did not struggle to track the speckle pattern in the way that OFM did. OFM performed decent tracking of the checkerboard pattern, but even for this pattern the NCC results more fully captured the motion of the target.

To quantify the difference between the NCC and OFM results, a point by point comparison was conducted. The difference between the displacement measured at every point by each method was recorded. Using this set of differences, an average correlation between the set of displacements was computed for each test. Average percent correlation results between the NCC and OFM displacements in both the horizontal and vertical directions are presented in Table 5.2. When calculating the correlation coefficient presented, a hypothesis test was performed to determine the significance of the correlation. The result of the hypothesis test was a p-value, with lower p-values (e.g. 0.001) implying more significant correlation. Some tests had high p-values which implied that the correlation was not significant. Comparisons that led to high p-values, implying insignificant correlation, were removed from the comparison if the p-value was greater than 0.2. Raw correlation results are presented along with the correlations after this removal, which served to remove some of the outliers. Overall, the results show that the checkerboard pattern resulted in a more consistent similarity of tracking between OFM and NCC, essentially doubling the result

from the speckle pattern. It is to be noted that there is a difference between correlation for horizontal and vertical displacement. The tests in this study only consisted of horizontal motions. NCC consistently measured zero change in vertical motion, while some vertical motion was measured by OFM, which can be used to quantify the accuracy resolution of the processing. Because NCC consistently measured zero change in vertical motion, but OFM did not, this led to a lower correlation between the two results. Overall, these results show that the correlation between the methods can be improved, which will be explored through discussion of several factors observed in this testing.

Table 5.2. Feasibility test: Average percent correlation between NCC and OFM displacement results.

<b>Target Pattern</b>	<b>Horizontal (%)</b>	<b>Horizontal (p &lt; 0.2) (%)</b>	<b>Vertical (%)</b>	<b>Vertical (p &lt; 0.2) (%)</b>
Speckle	38.7	42.8	14.3	27.6
Checkerboard	62.8	77.0	40.6	75.0

The results from this testing, which can be represented by the sample plots provided, pointed to two main areas of concern in the setup: rate of camera acquisition and lighting. The conditions of the image acquisition affected the robustness of the displacement tracking of the system. It was observed in both the OFM and NCC processing results that the camera frame rate defined in the code was not fast enough to fully capture the 3.0 Hz movement, leaving the displacement plots to appear jagged and at times fail to capture the extent of motion of the target. The 0.1 Hz rate was slow enough to capture the entire sinusoidal movement of the object and the 1.0 Hz was enough to create an acceptable sequence of the motion. Per the Nyquist frequency criterion mentioned in Section 3.3.3, a minimum frame rate of 6 fps was required to capture 3.0 Hz motion. As the rate of 10 fps proved insufficient for the desired resolution, a larger multiplication factor between frame rate and motion frequency is recommended to fully capture the motion of the target.

The initial tests revealed some constraints regarding the ideal lighting condition for the setup. Initially, it was assumed that, due to the inclusion of an LED light on the target card in the setup, the dark tests would perform well. However, the camera acquisition was affected by the darkness of the room, even though the target was well lit, so every few images appeared completely dark. Some flickering of the LED occurred that created this inconsistency. This severely affected

the results because in some of the tests points on the LED were tracked by OFM. When the light flickered, the points were lost completely, leaving holes in the tracking. This affected the NCC results as well, as the entire image would appear darker than the template, making correlation difficult. Thus, further testing was conducted to find suitable camera settings given this knowledge about the influence of lighting conditions.

### 5.1.2 Lighting Study Results

A lighting study was conducted to evaluate the effect of image brightness on processing. The results presented focus on visual inspection of the optical flow tracking and correlation of displacement results between the two image processing algorithms presented previously using this new set of images.

The beginning of this study involved an exploration of the aperture settings of the Tamron 672D lens. Sample images at six of the seven aperture levels, under both ambient lighting and a dark room, are illustrated in Figure 5.5 and Figure 5.6, respectively. The figures show the progression of image visibility from  $f/5.6$  (left) to  $f/32$  (right). Mild aperture settings of  $f/8$  (second from left) and  $f/16$  (fourth from left) were selected for further evaluation. The feasibility tests were all taken at  $f/8$ , so the results of the tests presented here are under the darker  $f/16$  for a comparison between the two.

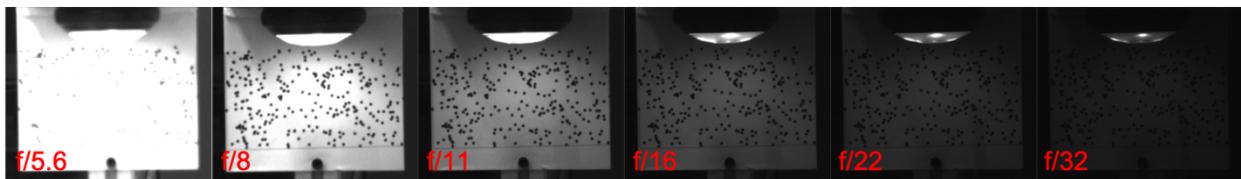


Figure 5.5. Various aperture settings under ambient lighting.

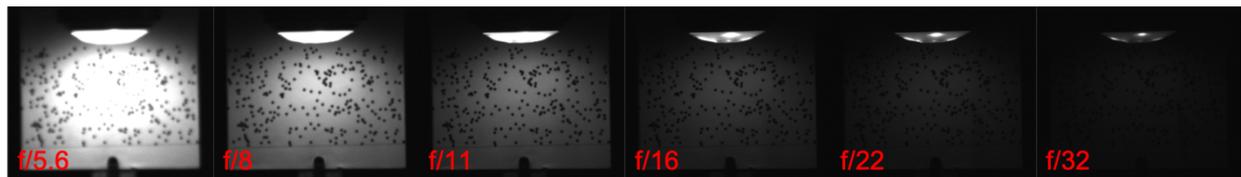


Figure 5.6. Various aperture settings under dark conditions.

In the initial phase of testing, images were originally taken with a lens at  $f/8$  and exhibited variability in the lighting due to flickering of the LED light. The setting of  $f/8$  was intentionally

bright for low-light conditions, such as previous field testing underneath a bridge at night. This setting combined with the flickering decreased the quality of the image processing. New images taken with a lens at  $f/16$  proved to be more consistent. Even though the images were darker and appeared poor to the human eye, they were processed more accurately by the OFM algorithm due to the brightness constancy offered at this setting. It should be noted that the ideal aperture setting will vary based on the ambient lighting conditions, so sample images should be evaluated before testing is conducted in order to verify the appropriate setting. Middle range apertures ( $f/8$  or  $f/16$ ) will typically be appropriate. Examples of this tracking for both targets are shown in Figure 5.7. This new set of images proved the inadequacy of the speckle pattern for OFM tracking, no matter the lighting, as it struggled to track selected feature points between images. This misidentification of feature points again led to erratic tracking. Tracking of the checkerboard pattern was more consistent. In the sample image shown, two feature points at the center intersection of the target and one point on the LED were tracked.



Figure 5.7. Lighting test: Sample optical flow tracking for speckle and checkerboard target patterns.

Processing by the NCC algorithm for this set of images included the selection of a smaller ROI to optimize the speed and accuracy tradeoff. The ROI was iterated five times for each target pattern, as illustrated in Figure 5.8. Regions of distinct patterns were chosen to be as small as possible while minimizing the accuracy tradeoff. For the speckle pattern, iteration #3 had the best tradeoff, as iterations #4 and #5 lost accuracy because the regions of speckles were less distinct when compared across the rest of the image. For the checkerboard pattern, iteration #5 provided

the best tradeoff as the smallest unique trackable region. For the speckle pattern, the largest ROI (#1) was processed at 0.72 images/sec and the chosen ROI (#3) was processed at 1.07 images/sec. For the checkerboard pattern, the largest ROI (#1) was processed at 0.63 images/sec and the chosen ROI (#5) was processed at 1.56 images/sec. On average, selecting the smaller ROI doubled the speed of the NCC processing. The OFM processing rate was 24.1 images/sec for the speckle pattern and 25.8 images/sec for the checkerboard, resulting in an overall average processing speed that was 19x faster than NCC, which is illustrated in Figure 5.9. The smaller ROIs optimized NCC over previous processing speeds, but it was still much slower than OFM. The ability of the OFM algorithm to process images around 25 fps under an image acquisition rate of 10 fps shows the potential for real-time processing of the system, as images can be processed more quickly than they are acquired.

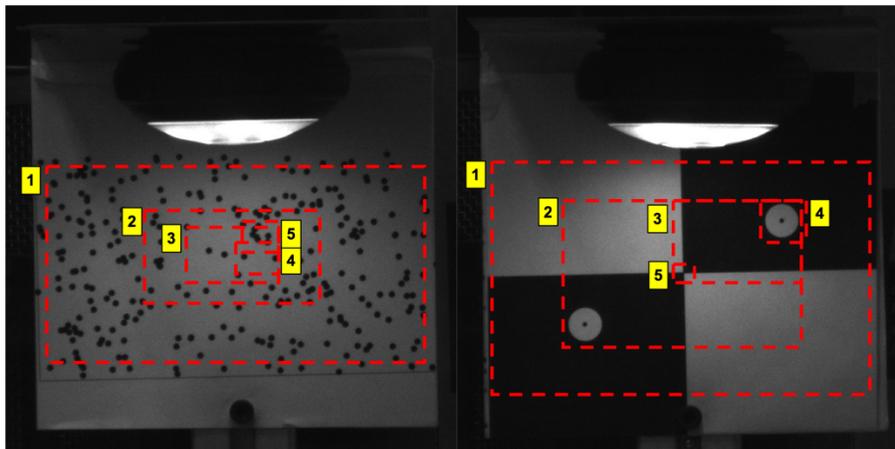


Figure 5.8. ROI iteration for speckle and checkerboard target patterns.

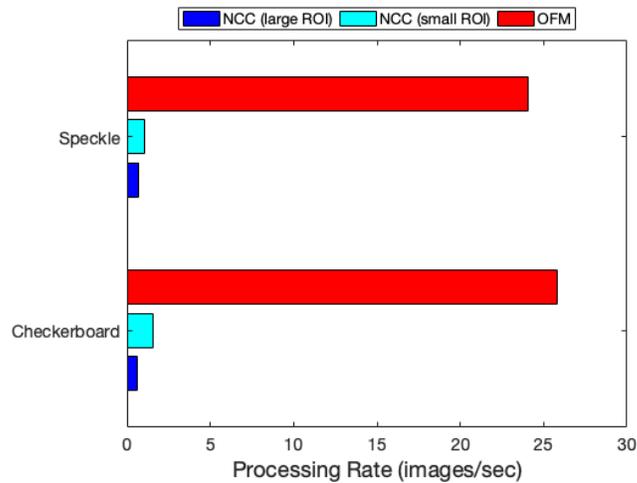


Figure 5.9. Average processing rates for speckle and checkerboard target patterns.

Sample horizontal displacement plots of the NCC and OFM results for both targets are shown in Figure 5.10 and Figure 5.11. The displacement was scaled from pixels to inches using the known dimensions of the target card (6 in. horizontal). The target motion was not sinusoidal like the previous tests, as the targets were instead translated across the length of the translation stage at a constant rate. Overall, NCC tracked the motion of both targets well due to the higher quality images captured by these new tests. Thus, when OFM struggled to track the object, it pointed to issues with OFM rather than with the images. As Figure 5.10 shows, OFM failed to track the feature points for the speckle pattern across the images, again due to misidentification of the feature points between frames. For the checkerboard pattern, OFM consistently tracked the linear horizontal displacement of the target and only exhibited a small margin of difference with the NCC displacement.

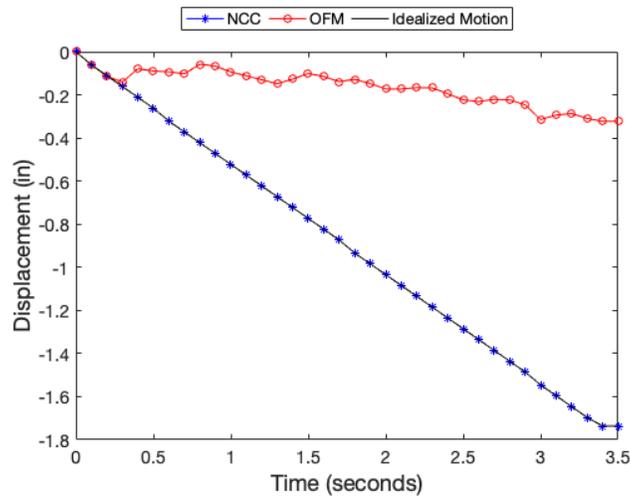


Figure 5.10. Lighting test: Sample horizontal displacement for speckle target pattern.

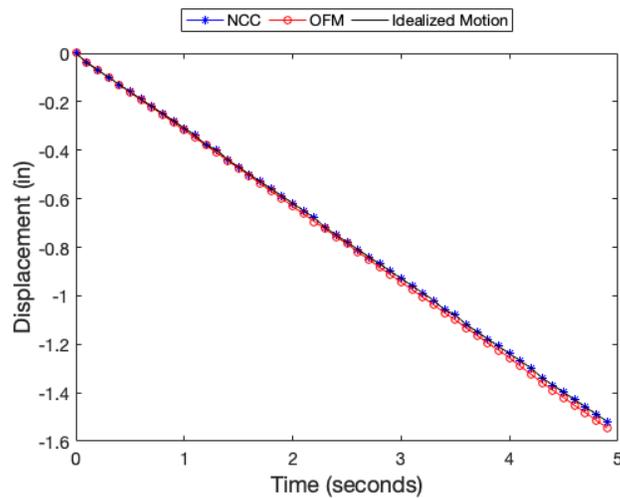


Figure 5.11. Lighting test: Sample horizontal displacement for checkerboard target pattern.

To quantify the difference between the NCC and OFM results, a point by point comparison was again conducted. The difference between the displacement measured at every point by each method was recorded. Using this set of differences, an average correlation between the set of displacements was computed for each test. The average percent correlation results between NCC and OFM for this new set of images are presented in Table 5.3. The results show that correlation was stronger for the checkerboard pattern, though that of the speckle pattern increased from the previous tests. The p-values were very low for all tests, implying significant correlation results. Again, there appears to be a difference of about 10% between the horizontal and vertical

correlation for each target, the reasons for which were discussed in the previous section. Because the images were of more consistent quality due to the change in aperture setting, these new results give a more accurate view of what the true correlation between methods is for each target. The high correlation between NCC and OFM confirms both the images and the algorithms. Thus, the results from this study confirm the parameters chosen from the previous testing for optimization of the OFM algorithm and the ability of OFM to track the movement of the checkerboard target pattern.

Table 5.3. Lighting test: Average percent correlation between NCC and OFM displacement results.

<b>Target Pattern</b>	<b>Horizontal (%)</b>	<b>Horizontal (p &lt; 0.2) (%)</b>	<b>Vertical (%)</b>	<b>Vertical (p &lt; 0.2) (%)</b>
Speckle	79.7	79.7	68.8	68.8
Checkerboard	99.3	99.3	90.3	90.3

### 5.1.3 Natural Feature Tracking Study Results

This study was conducted to evaluate the ability of the newly parametrized OFM algorithm to track natural features such as steel bolts or concrete cracks. Success would encourage further development of the algorithm for target-less purposes as a true feature tracking algorithm. The results are presented in the form of visual inspection of the behavior of the optical flow tracking and visual comparison of the displacement result across the targets.

Initial processing of the steel section and concrete cylinder test images found that the large field of view and background noise led to the tracking of stationary background features that were not associated with the moving target. Thus, all images were cropped to contain only the target object and a nominal margin around it. This included cropping of the LED light out of the images. In past target card-based testing, tracking of the LED light was acceptable because the LED was attached to the card and moved with the target. In these scenarios, however, the LED was stationary, so tracking of it (which occurred during some of the dark room tests) affected the accuracy of the displacement result. This created images of size 800 x 600 pixels for the side steel section, 700 x 380 pixels for the forward steel section, and 600 x 735 pixels for the concrete cylinder.

A sample graphic of the OFM tracking of each of the targets is shown in Figure 5.12. As mentioned, these targets were selected to evaluate the potential for tracking of the steel bolts and

the concrete crack. In the steel tests, the edges of the steel were tracked instead of the bolts. For the side view section, three of the four corners of the section were selected as feature points. Additionally, a point in the background was tracked in the sample result shown. For the forward view section, three points at the bottom intersection of the steel and the metal bar were selected as feature points. There was no discernable difference in the tracking results between the two steel orientations. For the concrete cylinder, five points adjacent to the crack were selected as feature points. The concrete tracking shows linear motion, but also experienced some variation due to misidentification of feature points across frames. In other tests, the crack itself served as a point of contrast and was tracked in addition to the edge of the cylinder or points in the background. For all targets, the dark lighting tests experienced the most consistent tracking because they removed background noise. Among each of the targets, the specific points that were selected for feature tracking were all due to the level of contrast in those regions, as was previously discussed. A significant factor that affected this level of contrast was the illumination of the target. In future applications, it is recommended that primary points of interest are individually illuminated in order to manipulate the contrast level. If future testing requires explicit tracking of features such as bolts or cracks, machine learning capabilities would need to be applied to the processing algorithm to train it to track such features specifically.

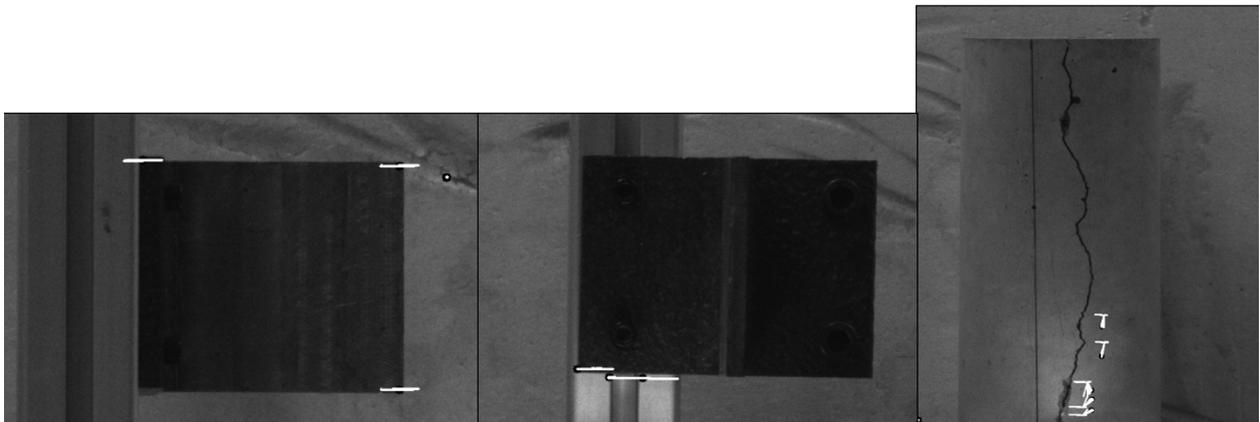


Figure 5.12. Natural feature test: Sample optical flow tracking of steel and concrete targets.

Processing rates for OFM were similar to previous tests, with vast improvements for the cropped images due to the decrease in image size. The side steel images had an average processing speed of 23.4 images/sec for the full image and 87.4 images/sec for the cropped image. The forward steel images had an average processing speed of 24.6 images/sec for the full image and

161.4 images/sec for the cropped image. The concrete images had an average processing speed of 24.5 images/sec for the full image and 89.7 images/sec for the cropped image.

Sample displacement results for each of the targets along with the idealized motion of the translation stage are shown in Figure 5.13. The displacement results were scaled from pixels to physical units using the known dimensions of the targets. The results shown here serve as a general representation of the results from the collection of tests across the variations in speed and lighting. The measured displacement of the forward steel section aligned with the shape of the translation stage motion, only experiencing a small amount of displacement error when measuring the peak magnitude. The side steel section exhibited the appropriate shape, but the magnitude of the displacement was off. This was due to the tracking of a stationary point in the background during this specific test, so the displacement average included that, which had the effect of scaling down the displacement result. Cropping of the test images sought to mitigate this issue by removing most of the background noise; however, not all background points could be completely removed because a nominal margin around the target itself was required that was at least equal to the magnitude of the expected motion. In the concrete displacement result, the feature points tracked measured the motion for about half of the test and then were misidentified during tracking when the motion reached its peak. Overall, this test proved that the system was capable of tracking more than just designed target cards by exhibiting natural feature tracking potential, given appropriate lighting conditions and distinct features that create points of high contrast on the desired target.

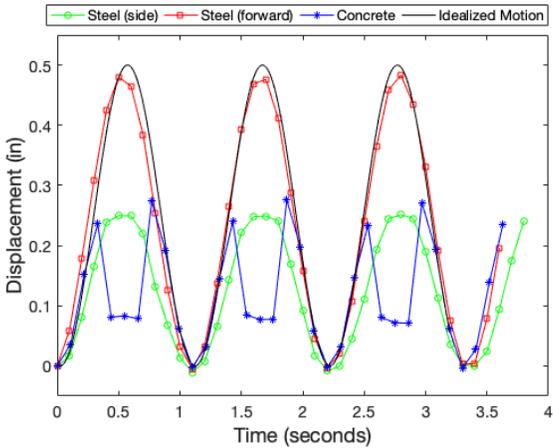


Figure 5.13. Natural feature test: Sample horizontal displacement for steel and concrete targets.

#### 5.1.4 Comparison Against Traditional Displacement Measurement Results

Testing was conducted to obtain a direct comparison of the proposed vision-based system against a traditional displacement measurement device. A cantilever frame was manually excited primarily in the horizontal direction, but was not fixed against vertical displacement. Results are presented in the form of visual inspection of the optical flow tracking and comparison between the displacement magnitudes found by each system.

Post-processing of the images using OFM measured displacement both horizontally and vertically. Initial processing of the images found that the field of view was too wide, leading to tracking of stationary high-contrast points on the adjacent string pot. Thus, images were cropped to a size of 1200 x 800 pixels to contain only the target card for more accurate displacement tracking. The full-size images (1920 x 1200 pixels) were processed at an average rate of 26.7 images/sec. The cropped images (1200 x 800 pixels) were processed twice as fast at a rate of 60.3 images/sec, which was expected given that this size was a fraction of that of the original image.

Sample optical flow tracking between the initial image and the final image of one of the tests is shown in Figure 5.14. The flow of the feature points due to the motion of the cantilever over time can be visualized. Based on the given lighting conditions, the selected features for tracking included three points on the checkerboard pattern (two points at the center and the middle of one of the white circles). The tracking of these three points on the card validated the design of the pattern for use in this feature point tracking application. Outside of the target card, four points on the light and two on the metal bracket holding the string pot to the end of the frame were tracked. The image could be cropped further to avoid tracking of points outside the target card; however, since the features tracked all moved uniformly as a rigid body, the tracking of this particular set of points did not affect the accuracy of the results. Points on the LED light were commonly selected as features in these tests because the brightness of the LED offered a high level of contrast against surrounding regions. The test images under darker conditions (dark room and f/16 aperture setting) only consisted of tracked points on the LED. As shown through the sample image, lighter settings allowed for the tracking of other features as well. This difference, however, neither proved nor disproved the accuracy of either condition, since the specific points to be tracked under either setting vary based on the context of the background. For example, though less points on the target card were tracked, the darker tests created more consistent images due to the removal of some of the background features, including the hand motion of the cantilever frame operator.



Figure 5.14. String pot comparison test: Sample optical flow tracking between initial and final image.

Results from the OFM processing were scaled from pixels to physical displacement using the known dimensions of the target card (6 in. horizontal). Both displacement measurements for one of the tests are plotted together in Figure 5.15. The string pot measured the horizontal movement at the tip of the cantilever where it was connected. The erratic motion of the cantilever was tracked well by the string pot at its high data acquisition rate. It can be seen that the OFM result followed the shape of the string pot displacement, but the OFM displacement is sharper due to a lower acquisition frequency than the string pot. Additionally, it can be noticed that the OFM displacement at peaks does not consistently equal that measured by the string pot.

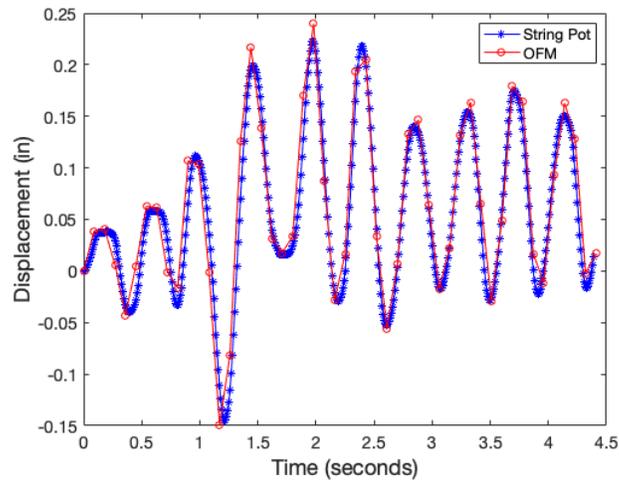


Figure 5.15. String pot comparison test: Sample string pot and OFM horizontal displacement.

A point by point correlation between the measurement results was not compiled because of differences in sampling rate between the devices. Instead, the ability of each measurement set to capture the magnitude of the local maximum and minimum displacements was evaluated. In the displacements shown in Figure 5.15, for example, the average difference in magnitude between the maximum displacements was  $-0.0058$  in. and that of the minimum displacements was  $0.0031$  in. Visually, this can be seen in that the OFM displacement matched with the string pot displacement at the negative peaks more consistently than it did with the positive peaks. This pointed to a trend of OFM displacement error being on the order of at least  $0.0058$  in., if the string pot measurements are considered a ground truth reference of the studied motion. Additionally, it was observed that there was no significant difference in image processing quality between the aperture settings or room lighting conditions. Frequency of motion was maintained as the primary factor affecting accuracy of OFM displacements and its ability to capture the maximum or minimum displacement magnitudes. However, the use of OFM provided an advantage for this type of displacement measurement when compared against the string pot as the string pot was only capable of uniaxial measurements, while OFM was able to track both horizontal and vertical motion. Overall, this study proved the applicability of OFM as a displacement measurement device while noting the accuracy of the method, which will be particularly useful in scenarios when use of traditional measurement sources like a string pot may not be feasible.

## 5.2 Field Testing Results

Field testing data from the testing scenario described in Section 4.4 was processed and the results are presented in the following section.

### 5.2.1 Traffic Signal Structure Deflection Study Results

The vision-based system was applied to track ambient wind excitations of a traffic signal structure without a reference of traditional sensors. All GS3 images from the six tests were post-processed using the OFM algorithm to determine the end displacement of the structure over time. The entire image was processed (no cropping) because it contained a sufficient field of view with minimal background noise and decent resolution of the target features. The OFM processing occurred at an average rate of 29.5 images/sec for all tests, which is about 7 seconds for a set of 200 images of size 1920 x 1200 pixels. Overall, the results show that the system was able to capture movement on a subpixel level. Depending on the specific level of contrast in the image, 13 to 16 points on a traffic sign on the mast arm were tracked (see Figure 5.16).

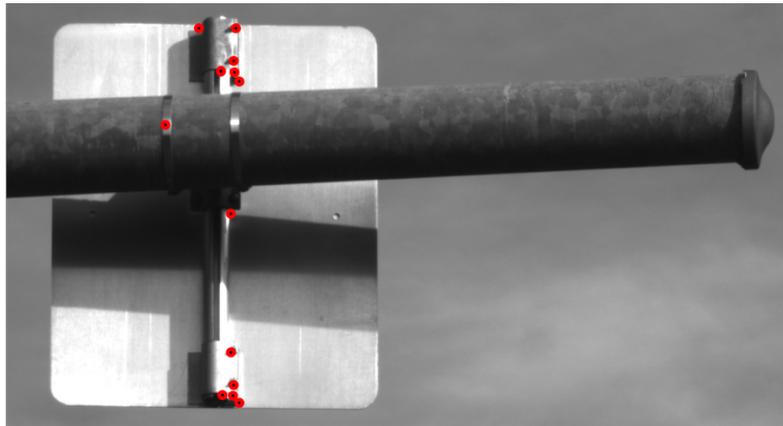


Figure 5.16. Traffic signal structure test: Sample optical flow feature point selection.

Pixel displacements were scaled to inches using a scale factor calculated from the known dimensions of the traffic sign (12 in. horizontal). Under ambient wind conditions, the cantilevered mast arm exhibited harmonic motion with peak displacements above 0.015 in., as illustrated in Figure 5.17. Testing conditions noted minimal wind, so the displacement magnitude was reasonable. The scale that displacements were able to be captured at points to the resolution of the system due to the limited motion present. The extent of the horizontal movement, in this specific

scenario, was considered the order of error of the system (level of accuracy), as horizontal motion of the structure was expected to be relatively fixed. In the test shown in Figure 5.17, this constituted an accuracy of 0.005 in., which was consistent with the accuracy presented in the string pot displacement comparison study. The rest of the data from the other five tests is graphically presented in the Appendix Figures A1 to A6.

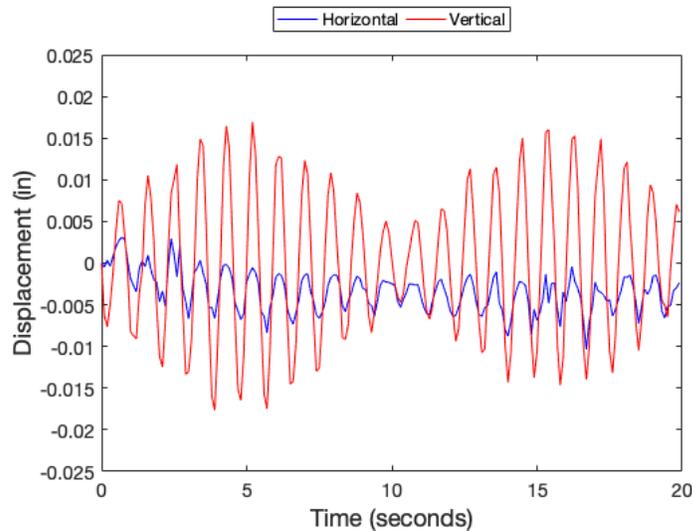


Figure 5.17. Traffic signal structure test: Sample OFM horizontal and vertical displacement.

This field test proved the robustness of the OFM algorithm under uncontrolled field conditions in its ability to capture small, damped oscillations of the traffic signal structure. OFM performed well with minor background noise such as a bird flying and the clouds moving. The 20-minute late afternoon testing period optimized the consistency of the lighting, but even within this, small variations were apparent in the image processing results. Slightly different points were tracked between some of the tests, based on the specific variations of light during the time of the test. The first test used an aperture setting of  $f/8$  which was changed to  $f/16$  for subsequent tests based on the level of brightness of the ambient light. The darker aperture setting created a more favorable level of contrast in the later images. Finally, it should be noted that displacement presented was not corrected due to the eccentricity between the structure and the camera. The displacement calculation uses point-based measurements which, opposed to area-based measurements, are independent of each other. Therefore, the level of angle of the sign was assumed

to have a minimal effect on the displacement tracking of the rigid body. As mentioned, the GoPro images were acquired as a secondary source and the processing results are not presented here.

Because there was no target pattern, these test images were not optimized for the use of the NCC algorithm. A comparison was not be compiled; however, one set of images was processed using NCC for baseline validity. The NCC algorithm was able to capture the magnitude of the displacement but did a poor job of tracking the oscillatory motion of the sign. On top of that, the NCC computation speed extremely lagged behind that of OFM, as it processed 0.35 images/sec, meaning that the OFM processing was 84x faster for this set of images. Thus, the development of the OFM algorithm and processing onboard the Jetson created a great opportunity to test the system by tracking traffic signal structure displacement, which could not have been done using previous methods alone.

## 6 Conclusion

This work has proven the feasibility of a contactless vision-based system for structural displacement measurement with real-time processing potential. The work developed here has led to the creation of a vision-based system that can:

- (1) Monitor a structure in a complex environment that cannot be reached for attachment of traditional sensors, such as a traffic signal structure, and obtain subpixel displacement measurements of low frequency structural motions under variable lighting conditions.
- (2) Visually track a structure's natural features using optical flow with accuracy on the order of 0.005 in. so that manmade target cards do not need to be mounted.
- (3) Process images quickly by using a feature-based method so that results can be reviewed before leaving the field site. Final results were not processed in real-time, but the algorithm proved to be very fast, so such a capability has the potential to become a part of the process.

The main application of this work is for short-term monitoring of bridges or traffic signal structures. Though it was not evaluated in the experimental studies presented here, the ability to instrument the system on the structure itself for monitoring when no fixed ground is available adjacent to the structure should be studied next as it creates opportunities for monitoring of bridges in complex environments. The application for localized girder deflection measurement for bridge load rating purposes is possible based on this proposed direction.

A main limitation of this system relates to its current computational capabilities. The image acquisition rate affected the sampling rate of the images and therefore the quality of the displacement measurements. Optimizing this rate will increase the ability of the system to work with a variety of motion frequencies and avoid aliasing. Another tradeoff is due to the image processing itself. The algorithm implemented proved to be quick (exhibiting potential for real-time processing) and suitable for capturing displacements due to low frequency motions under relatively controlled lighting conditions, but lacked robustness in the face of changes in these conditions. Overall, the current level of accuracy was acceptable based on the order of displacement typically seen in civil structure monitoring, but improvements can always be made. Thus, next steps for expanding this work will be discussed in the following section.

## 6.1 Future Work

Building off of the foundation of this initial work, this vision-based system has the potential for further development in order to extend its processing capabilities and applications for field deployment. Recommended steps for future work to enable deployment of the system in the built environment for short-term monitoring of bridges or traffic signal structures will be offered here. Further development of this work will be separated into two areas: physical requirements and computational capabilities.

### 6.1.1 *Physical Requirements*

In the testing conducted, the system was always plugged into a consistent power source. This took the form of a wall power outlet and a gas-powered generator for the lab and field testing, respectively. The ability of the system to be powered on its own or while connected to a remote battery will need to be studied. The Jetson TX2 is lauded for its efficient energy usage, which will need to be taken advantage of. The power consumption requirements of the system will determine the necessary battery extensions for field deployment and serve as a limiting factor for the duration of such deployment.

The system was also connected to a desktop monitor, which allowed for live visualization of the image acquisition (including camera focusing) and user control of the acquisition through a command line interface. During the field test of the traffic signal structure, this same setup was used, but it is expected that most field applications of this system would not allow for such a luxury. Thus, to prepare for circumstances in which the system is not able to be connected to a monitor directly, wireless methods for such control should be explored. The system could be connected via remote Wi-Fi to a tablet, for example, where the user can then verify the performance of the acquisition by viewing the live data feed. This was the control method for the GoPro camera introduced in this study. This level of connectivity allows the user to maintain vigilance of any issues in data acquisition almost instantaneously through this form of remote access. A system setup that is based on not having a visual interface for camera control would require a high level of computer vision capabilities, which will be referenced in the following subsection.

A main application of this system is for deployment under a bridge for collection of localized bridge deflection measurements when there is no adjacent fixed ground for camera

placement. Such a setup was used in previous testing to measure DIC-based displacements of a patterned target card that was mounted at the mid-span underside of a bridge girder (Alipour et al. 2019). The camera itself was mounted on a pier cap in order to measure girder deflections, as depicted in Figure 6.1. The instrumentation of a camera directly to a bridge structure, rather than on nearby fixed ground, creates some uncertainty regarding movement of the camera. Naturally, the camera will experience any rotation that the pier cap itself, though a rigid body, experiences. Vertical bridge deflection measurement using DIC has been validated (Peddle et al. 2011), but mounting the camera to the bridge in order to obtain such measurements has not. Thus, there is interest in quantifying this motion so that it can be subtracted out of the system for this sort of measurement. In previous testing, it was assumed that this movement was negligible or, at least, relative compared to the overall deflection of the structure under truck loading, but some results created uncertainty regarding this assumption.



Figure 6.1. Camera mounted on bridge pier cap.

As part of this work, a field test was performed to quantify pier cap movement to determine the amount of mounted camera movement and its effect on deflection measurements. The data was not usable due to inaccurate assumptions regarding the sensors chosen for measurements under dynamic loading, but the intent of the study is valuable and would contribute to future use of the system under bridges without fixed ground placement. Thus, the premise of that study will be presented here for future evaluation.

The Brattons Creek Bridge in Rockbridge County, Virginia, was selected for the study due to ground accessibility underneath one of its spans and because it was the subject of previous

load testing (Ndong 2018). This 3-span simply supported reinforced concrete bridge has a total length of 98 ft 2 in. with zero skew and is shown in Figure 6.2. Each of the spans is 32 ft long and the bridge is 23 ft 8 in. wide and spans Brattons Creek. The bridge is configured with four reinforced concrete longitudinal T-beams supporting an 8 in. reinforced concrete deck and supported by solid reinforced concrete piers and concrete abutments. The rectangular T-beams each have a stem width of 16 in. and height of 24 in.



Figure 6.2. Brattons Creek Bridge.

The bridge was instrumented per Figure 6.3. The instrumentation included cameras for DIC-based deflection measurements, string potentiometers (SP) for deflection measurements, and tiltmeters (T) for rotation measurements. Patterned target cards with speckles of size 2 mm were mounted to the underside of the beam at mid-span on each of the beams. Three GoPros were mounted on the right pier cap adjacent to each beam, one GoPro was mounted on the left pier cap adjacent to the middle beam, and two GoPros were placed on tripods next to beams 1 and 2. This setup allowed for comparison between the results of the cameras on fixed ground and the cameras mounted on the pier caps. These deflections would also be compared to the string pot-measured deflections using the tiltmeter data. One string pot was attached to each beam at mid-span and one was placed on an exterior location of the pier cap. The string pots were supported by tripods and sat on a block-frame interface. The three tiltmeters were mounted on the pier cap underneath each of the three beams. This setup followed the points of interest from the previous load testing, without the instrumentation of strain gauges or accelerometers.

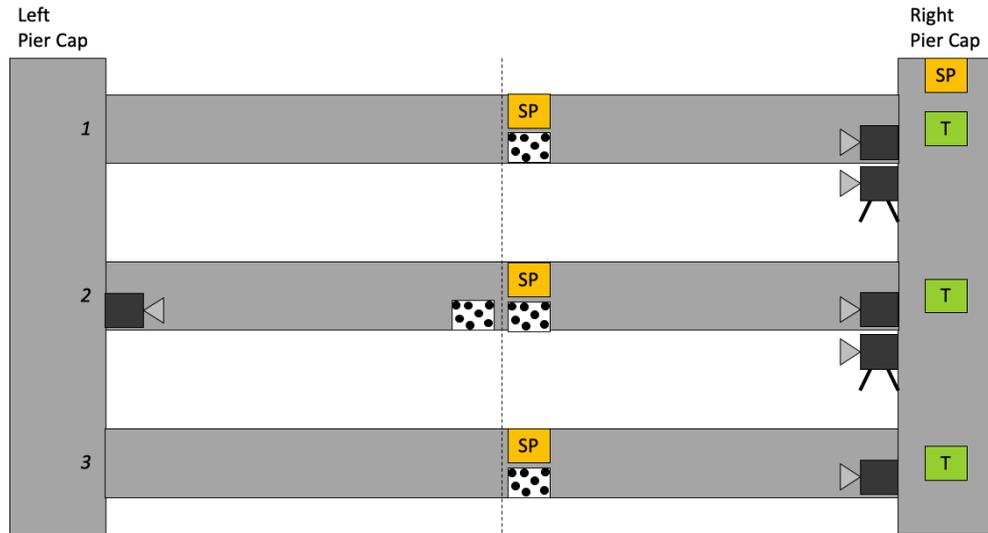


Figure 6.3. Pier cap study: Bridge instrumentation.

The goal of this was to collect data on how the pier cap moved under traffic loading in order to quantify its magnitude to decide if it was negligible or not and, if not, to include it in the deflection calculations. If not negligible, the rotation measured by the tiltmeters can be correlated with a deflection (as shown in Figure 6.4) and subtracted from the mounted camera's DIC measured deflection at mid-span to calibrate a final DIC-based deflection, per Equation 4.1. Ideally, this final DIC-based deflection would be equal to the ground truth deflection measured by the string pot and that of the camera on fixed ground, per Equation 4.2. The deflection measured by the camera placed on the tripod on fixed ground should equal this measurement as well.

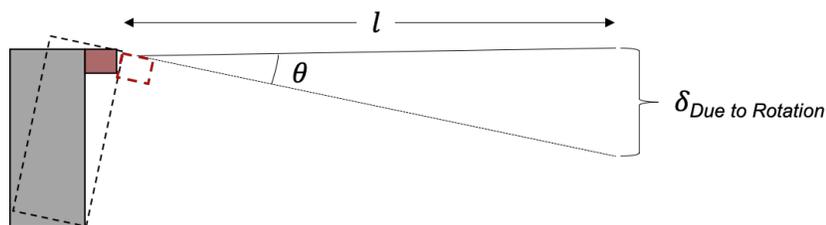


Figure 6.4. Deflection due to pier cap rotation.

$$\delta_{DIC \text{ Final}} = \delta_{DIC \text{ Mounted}} - \delta_{\text{Due to Rotation}} \quad (4.1)$$

$$\delta_{\text{String Pot Measured}} = \delta_{DIC \text{ Fixed}} = \delta_{DIC \text{ Final}} \quad (4.2)$$

In order to facilitate accurate measurement acquisition by the sensors, controlled static loading should occur with the load vehicle stopped on the bridge at the location of the tiltmeters, allowing the devices to settle, and then recording data. This is recommended because the tiltmeters are not suitable for dynamic testing due to a “fluid bubble” technology that calculates tilt after settlement. This requires settlement time for zeroing and thus increases the response time needed for use, thus limiting their application for dynamic testing (Bridge Diagnostics, Inc. 2012). Appropriate use of the tiltmeters under the instrumentation described will allow for quantification of pier cap movement and its effect on deflection measurements.

### *6.1.2 Computational Capabilities*

The NVIDIA Jetson TX2 brings top of the line computing capabilities, such as a GPU, that were hardly utilized in this preliminary work. Utilization of the GPU itself, in addition to the computer architecture of the TX2, creates many opportunities for real-time processing. The OFM processing speeds noted in this study provide a strong foundation for advancing the efficiency of the overall system processing.

The use of computer vision can be expanded to optimize several features of the proposed system. Machine learning can be used to automate the camera focus and image capture process. This would involve training the computer to create good quality images by knowing what the desired target is and what a good quality image consists of. The system could subsequently be designed to specifically track certain features such as steel bolts or concrete cracks, as was hinted at previously. The TX2 is highly capable of machine learning processes, which would further the development of the system to become autonomous and require less user control.

## References

- Adeli, H., and Kamal, O. (1993). *Parallel Processing in Structural Engineering*. Elsevier Science Publishers, Essex, England.
- Alahi, A., Ortiz, R., and Vandergheynst, P. (2012). “FREAK: Fast Retina Keypoint.” *2012 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Providence, RI, 510–517.
- Alipour, M., Washlesky, S. J., and Harris, D. K. (2019). “Field Deployment and Laboratory Evaluation of 2D Digital Image Correlation for Deflection Sensing in Complex Environments.” *Journal of Bridge Engineering*, 24(4), 1–15.
- Almeida, E. de B. (2015). “Revisiting Normalized Cross-Correlation for Accurate Camera Pose Estimation and Accurate Real-Time Multiple View Stereo.” Brown University.
- Amert, T., Otterness, N., Yang, M., Anderson, J. H., and Smith, F. D. (2017). “GPU Scheduling on the NVIDIA TX2: Hidden Details Revealed.” *2017 IEEE Real-Time Systems Symposium (RTSS)*, IEEE, Paris, 104–115.
- Asano, S., Maruyama, T., and Yamaguchi, Y. (2009). “Performance comparison of FPGA, GPU and CPU in image processing.” *2009 International Conference on Field Programmable Logic and Applications*, IEEE, Prague, Czech Republic, 126–131.
- Baker, S., and Matthews, I. (2001). “Equivalence and efficiency of image alignment algorithms.” *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, IEEE Comput. Soc, Kauai, HI, USA, I-1090-I-1097.
- Barbacci, A., Diener, J., Hémon, P., Adam, B., Donès, N., Reveret, L., and Moulia, B. (2014). “A robust videogrametric method for the velocimetry of wind-induced motion in trees.” *Agricultural and Forest Meteorology*, 184, 220–229.
- Bar-Kochba, E., Toyjanova, J., Andrews, E., Kim, K.-S., and Franck, C. (2015). “A Fast Iterative Digital Volume Correlation Algorithm for Large Deformations.” *Experimental Mechanics*, 55(1), 261–274.
- Bartilson, D. T., Wieghaus, K. T., and Hurlebaus, S. (2015). “Target-less computer vision for traffic signal structure vibration studies.” *Mechanical Systems and Signal Processing*, 60–61, 571–582.
- Bartoli, G., Facchini, L., Pieraccini, M., Fratini, M., and Atzeni, C. (2008). “Experimental utilization of interferometric radar techniques for structural monitoring.” *Structural Control and Health Monitoring*, 15(3), 283–298.

- Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). “Speeded-Up Robust Features (SURF).” *Computer Vision and Image Understanding*, 110(3), 346–359.
- Betke, M., Haritaoglu, E., and Davis, L. S. (2000). “Real-time multiple vehicle detection and tracking from a moving vehicle.” *Machine Vision and Applications*, 12(2), 69–83.
- Blaber, J., Adair, B., and Antoniou, A. (2015). “Ncorr: Open-Source 2D Digital Image Correlation Matlab Software.” *Experimental Mechanics*, 55(6), 1105–1122.
- Bouguet, J.-Y. (2001). “Pyramidal implementation of the affine lucas kanade feature tracker: Description of the algorithm.” Intel Corporation.
- Bridge Diagnostics, Inc. (2012). “Tiltmeter Operations Manual.”
- Brooks, C., Dobson, R., Banach, D., Oommen, T., Mukherjee, A., Havens, T., Ahborn, T., Bhat, C., Zhao, S., Lyu, Q., and Marion, N. (2018). *Implementation of Unmanned Aerial Vehicles (UAVs) for Assessment of Transportation Infrastructure - Phase II*. Michigan Department of Transportation, 164.
- Brown, R. J., Gaebler, K. O., Shield, C. K., and Linderman, L. E. (2019). *Displacement Monitoring of I-35W Saint Anthony Falls Bridge with Current Vibration-Based System*. Minnesota Department of Transportation, 72.
- Bruck, H. A., McNeill, S. R., Sutton, M. A., and Peters, W. H. (1989). “Digital image correlation using newton-raphson method of partial differential correction.” *Experimental Mechanics*, 29(3), 261–267.
- Busca, G., Cigada, A., Mazzoleni, P., and Zappa, E. (2014). “Vibration Monitoring of Multiple Bridge Points by Means of a Unique Vision-Based Measuring System.” *Experimental Mechanics*, 54(2), 255–271.
- Caetano, E., Silva, S., and Bateira, J. (2011). “A Vision System for Vibration Monitoring of Civil Engineering Structures.” *Experimental Techniques*, 35(4), 74–82.
- Calonder, M., Lepetit, V., Strecha, C., and Fua, P. (2010). “BRIEF: Binary Robust Independent Elementary Features.” *Computer Vision – ECCV 2010*, Springer Berlin Heidelberg, Berlin, Heidelberg, 778–792.
- Catbas, F. N., Zaurin, R., Gul, M., and Gokce, H. B. (2012). “Sensor Networks, Computer Imaging, and Unit Influence Lines for Structural Health Monitoring: Case Study for Bridge Load Rating.” *Journal of Bridge Engineering*, 17(4), 662–670.

- Chiang, C.-H., Shih, M.-H., Chen, W., and Yu, C.-P. (2011). “Displacement measurements of highway bridges using digital image correlation methods.” *7th International Symposium on Precision Engineering Measurements and Instrumentation*, K.-C. Fan, M. Song, and R.-S. Lu, eds., Yunnan, China, 83211G.
- Dilshan, N. (2017). *Feature Tracker*.
- Dworakowski, Z., Kohut, P., Gallina, A., Holak, K., and Uhl, T. (2016). “Vision-based algorithms for damage detection and localization in structural health monitoring: Vision-based Algorithms for Damage Detection and Localization.” *Structural Control and Health Monitoring*, 23(1), 35–50.
- Eberl, C., Gianola, D. S., and Bundschuh, S. (2010). *Digital Image Correlation and Tracking*.
- Elliott, G. A., and Anderson, J. H. (2011). “Real-World Constraints of GPUs in Real-Time Systems.” *2011 IEEE 17th International Conference on Embedded and Real-Time Computing Systems and Applications*, IEEE, Toyama, Japan, 48–54.
- Feng, D., Feng, M., Ozer, E., and Fukuda, Y. (2015a). “A Vision-Based Sensor for Noncontact Structural Displacement Measurement.” *Sensors*, 15(7), 16557–16575.
- Feng, D., and Feng, M. Q. (2017). “Experimental validation of cost-effective vision-based structural health monitoring.” *Mechanical Systems and Signal Processing*, 88, 199–211.
- Feng, D., and Feng, M. Q. (2018). “Computer vision for SHM of civil infrastructure: From dynamic response measurement to damage detection – A review.” *Engineering Structures*, 156, 105–117.
- Feng, M. Q., Fukuda, Y., Feng, D., and Mizuta, M. (2015b). “Nontarget Vision Sensor for Remote Measurement of Bridge Dynamic Response.” *Journal of Bridge Engineering*, 20(12), 04015023.
- Fukuda, Y., Feng, M. Q., Narita, Y., Kaneko, S., and Tanaka, T. (2013). “Vision-Based Displacement Sensor for Monitoring Dynamic Response Using Robust Object Search Algorithm.” *IEEE Sensors Journal*, 13(12), 4725–4732.
- Fukuda, Y., Feng, M. Q., and Shinozuka, M. (2010). “Cost-effective vision-based system for monitoring dynamic response of civil engineering structures.” *Structural Control and Health Monitoring*, 17(8), 918–936.
- Google Maps. (2019). “Fontaine Ave at Ray C Hunt Dr.” <<https://www.google.com/maps/@38.0264862,-78.5246748,38m/data=!3m1!1e3>>.

- Guan, S., Bridge, J. A., Li, C., and DeMello, N. J. (2019). "Smart Radar Sensor Network for Bridge Displacement Monitoring." *J. Bridge Eng.*, 24(1), 11.
- Guizar-Sicairos, M., Thurman, S. T., and Fienup, J. R. (2008). "Efficient subpixel image registration algorithms." *Optics Letters*, 33(2), 156.
- Harris, C., and Stephens, M. (1988). "A Combined Corner and Edge Detector." *Proceedings of the Alvey Vision Conference 1988*, Alvey Vision Club, Manchester, 23.1-23.6.
- Hellier, C. J. (2003). *Handbook of Nondestructive Evaluation*. The McGraw-Hill Companies.
- Horn, B., and Shunk, B. (1981). "Determining optical flow." *Artificial Intelligence*, 17, 185–203.
- Idzenga, T., Gaburov, E., Vermin, W., Menssen, J., and De Korte, C. L. (2014). "Fast 2-D ultrasound strain imaging: the benefits of using a GPU." *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 61(1), 207–213.
- Im, S. B., Hurlebaus, S., and Kang, Y. J. (2013). "Summary Review of GPS Technology for Structural Health Monitoring." *Journal of Structural Engineering*, 139(10), 1653–1664.
- Infrastructure Report Card*. (2017). American Society of Civil Engineers, 1–110.
- Intel Corporation. (2019). *OpenCV*. Intel Corporation.
- Jáuregui, D. V., White, K. R., Woodward, C. B., and Leitch, K. R. (2003). "Noncontact Photogrammetric Measurement of Vertical Bridge Deflection." *Journal of Bridge Engineering*, 8(4), 212–222.
- Ji, Y. F., and Chang, C. C. (2008). "Nontarget Image-Based Technique for Small Cable Vibration Measurement." *Journal of Bridge Engineering*, 13(1), 34–42.
- Kim, D.-H., and Feng, M. Q. (2007). "Real-Time Structural Health Monitoring Using a Novel Fiber-Optic Accelerometer System." *IEEE Sensors Journal*, 7(4), 536–543.
- Kim, J.-S., Hwangbo, M., and Kanade, T. (2009). "Realtime affine-photometric KLT feature tracker on GPU in CUDA framework." *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, IEEE, Kyoto, 886–893.
- Kong, X., and Li, J. (2018). "Automated fatigue crack identification through motion tracking in a video stream." *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2018*, H. Sohn, ed., SPIE, Denver, United States, 29.
- Lee, J., Lee, K.-C., Cho, S., and Sim, S.-H. (2017). "Computer Vision-Based Structural Displacement Measurement Robust to Light-Induced Image Degradation for In-Service Bridges." *Sensors*, 17(10), 2317.

- Leutenegger, S., Chli, M., and Siegwart, R. Y. (2011). “BRISK: Binary Robust invariant scalable keypoints.” *2011 International Conference on Computer Vision*, IEEE, Barcelona, Spain, 2548–2555.
- Lewis, J. P. (1995). “Fast Normalized Cross-Correlation.” 1–7.
- Lim, S., Apostolopoulos, J. G., and Gamal, A. E. (2005). “Optical flow estimation using temporally oversampled video.” *IEEE Transactions on Image Processing*, 14(8), 1074–1087.
- Lowe, D. G. (2004). “Distinctive Image Features from Scale-Invariant Keypoints.” *International Journal of Computer Vision*, 60(2), 91–110.
- Lucas, B. D., and Kanade, T. (1981). “An Iterative Image Registration Technique with an Application to Stereo Vision.” *7th International Joint Conference on Artificial Intelligence (IJCAI)*, Vancouver, British Columbia, 121–130.
- Luo, L., Feng, M. Q., and Wu, Z. Y. (2018). “Robust vision sensor for multi-point displacement monitoring of bridges in the field.” *Engineering Structures*, 163, 255–266.
- “Manual for Bridge Evaluation.” (2018). American Association of State Highway and Transportation Officials.
- Min, J.-H., Gelo, N. J., and Jo, H. (2015). “Non-contact and Real-time Dynamic Displacement Monitoring using Smartphone Technologies.” *Life Cycle Reliability and Safety Engineering*, (2), 12.
- Morlier, J., and Michon, G. (2010). “Virtual Vibration Measurement Using KLT Motion Tracking Algorithm.” *Journal of Dynamic Systems, Measurement, and Control*, 132(1), 011003.
- Nassif, H. H., Gindy, M., and Davis, J. (2005). “Comparison of laser Doppler vibrometer with contact sensors for monitoring bridge deflection and vibration.” *NDT & E International*, 38(3), 213–218.
- Ndong, A. K. (2018). “A Methodology for Condition Assessment of T-Beam Bridges without Structural Plans.” University of Virginia.
- NVIDIA Corporation. (2018). “CUDA C Programming Guide.”
- NVIDIA Corporation. (2019). “NVIDIA Jetson Systems.” *NVIDIA*, <<https://www.nvidia.com/en-us/autonomous-machines/embedded-systems-dev-kits-modules/>>.
- Ojio, T., Carey, C. H., O'Brien, E. J., Doherty, C., and Taylor, S. E. (2016). “Contactless Bridge Weigh-in-Motion.” *Journal of Bridge Engineering*, 21(7), 04016032.

- Otterness, N., Yang, M., Rust, S., Park, E., Anderson, J. H., Smith, F. D., Berg, A., and Wang, S. (2017). "An Evaluation of the NVIDIA TX1 for Supporting Real-Time Computer-Vision Workloads." *2017 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, IEEE, Pittsburg, PA, USA, 353–364.
- Pan, B., Asundi, A., Xie, H., and Gao, J. (2009a). "Digital image correlation using iterative least squares and pointwise least squares for displacement field and strain field measurements." *Optics and Lasers in Engineering*, 47(7–8), 865–874.
- Pan, B., Li, K., and Tong, W. (2013). "Fast, Robust and Accurate Digital Image Correlation Calculation Without Redundant Computations." *Experimental Mechanics*, 53(7), 1277–1289.
- Pan, B., Qian, K., Xie, H., and Asundi, A. (2009b). "Two-dimensional digital image correlation for in-plane displacement and strain measurement: a review." *Measurement Science and Technology*, 20(6), 062001.
- Pan, B., Tian, L., and Song, X. (2016). "Real-time, non-contact and targetless measurement of vertical deflection of bridges using off-axis digital image correlation." *NDT & E International*, 79, 73–80.
- Park, K., Torbol, M., and Kim, S. (2018). "Vision-Based Natural Frequency Identification Using Laser Speckle Imaging and Parallel Computing: Vision-based natural frequency identification using laser speckle imaging and parallel computing." *Computer-Aided Civil and Infrastructure Engineering*, 33(1), 51–63.
- Peddle, J., Goudreau, A., Carlson, E., and Santini-Bell, E. (2011). "Bridge displacement measurement through digital image correlation." *Bridge Structures*, (4), 165–173.
- Peddle, J. T. (2011). "Digital image correlation as a tool for bridge load rating and long-term evaluation." University of New Hampshire.
- Reagan, D., Sabato, A., and Niezrecki, C. (2017). "Unmanned aerial vehicle acquisition of three-dimensional digital image correlation measurements for structural health monitoring of bridges." *SPIE Smart Structures and Materials + Nondestructive Evaluation and Health Monitoring*, H. F. Wu, A. L. Gyekenyesi, P. J. Shull, and T.-Y. Yu, eds., Portland, Oregon, United States, 1016909.

- Ribeiro, D., Calçada, R., Ferreira, J., and Martins, T. (2014). “Non-contact measurement of the dynamic displacement of railway bridges using an advanced video-based system.” *Engineering Structures*, 75, 164–180.
- Rice, J. A., Gu, C., Li, C., and Guan, S. (2012). “A radar-based sensor network for bridge displacement measurements.” *SPIE Smart Structures and Materials + Nondestructive Evaluation and Health Monitoring*, M. Tomizuka, C.-B. Yun, and J. P. Lynch, eds., San Diego, California, 83450I.
- Rosten, E., and Drummond, T. (2006). “Machine Learning for High-Speed Corner Detection.” *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, eds., Springer Berlin Heidelberg, Berlin, Heidelberg, 430–443.
- Seghir, R., Witz, J. F., and Coudert, S. (2014). *YaDICs - Digital Image Correlation 2/3D software*.
- Shafiei Dizaji, M., Alipour, M., and Harris, D. K. (2017). “Leveraging Vision for Structural Identification: A Digital Image Correlation Based Approach.” *International Digital Imaging Correlation Society*, M. Sutton and P. L. Reu, eds., Springer International Publishing, Cham, 121–124.
- Shariati, A., and Schumacher, T. (2017). “Eulerian-based virtual visual sensors to measure dynamic displacements of structures.” *Structural Control and Health Monitoring*, 24(10), e1977.
- Shi, J., and Tomasi, C. (1994). “Good features to track.” *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition CVPR-94*, IEEE Comput. Soc. Press, Seattle, WA, USA, 593–600.
- Smith, R. (2017). “NVIDIA Announces Jetson TX2: Parker Comes To NVIDIA’s Embedded System Kit.” *AnandTech*, <<https://www.anandtech.com/show/11185/nvidia-announces-jetson-tx2-parker>>.
- Song, Y.-Z., Bowen, C. R., Kim, A. H., Nassehi, A., Padgett, J., and Gathercole, N. (2014). “Virtual visual sensors and their application in structural health monitoring.” *Structural Health Monitoring: An International Journal*, 13(3), 251–264.
- Sotelino, E. D. (2003). “Parallel Processing Techniques in Structural Engineering Applications.” *Journal of Structural Engineering*, 129(12), 1698–1706.

- Sutton, M. A., Orteu, J.-J., and Schreier, H. W. (2009). *Image Correlation for Shape, Motion and Deformation Measurements: Basic Concepts, Theory and Applications*. Springer, New York.
- Taylor, S., and Lydon, M. (2017). “Advanced SHM using Computer Vision and Machine Learning.” *8th International Conference on Structural Health Monitoring of Intelligent Infrastructure*, Brisbane, Australia, 8.
- The MathWorks, Inc. (2019). *Matlab*. The MathWorks, Inc.
- Torbol, M. (2014). “Real-Time Frequency-Domain Decomposition for Structural Health Monitoring Using General-Purpose Graphic Processing Unit: Real-time frequency-domain decomposition for SHM using GPGPU.” *Computer-Aided Civil and Infrastructure Engineering*, 29(9), 689–702.
- Turner, D. Z. (2015). *Digital Image Correlation Engine (DICe) Reference Manual*. Sandia National Laboratory.
- Ullah, F., Kaneko, S., and Igarashi, S. (2001). “Orientation code matching for robust object search.” *IEICE Trans. Inf. Syst.*, 999–1006.
- WindAlert. (2019). “Charlottesville-Albemarle.” <<https://windalert.com/spot/17913>>.
- Wolfer, J. (2015). “A heterogeneous supercomputer model for high-performance parallel computing pedagogy.” *2015 IEEE Global Engineering Education Conference (EDUCON)*, IEEE, Tallinn, Estonia, 799–805.
- Xu, Y., Brownjohn, J., and Kong, D. (2018). “A non-contact vision-based system for multipoint displacement monitoring in a cable-stayed footbridge.” *Structural Control and Health Monitoring*, 25(5), 1–23.
- Xu, Y., Brownjohn, J. M. W., and Huseynov, F. (2019). “Accurate Deformation Monitoring on Bridge Structures Using a Cost-Effective Sensing System Combined with a Camera and Accelerometers: Case Study.” *Journal of Bridge Engineering*, 24(1), 1–14.
- Ye, X. W., Ni, Y. Q., Wai, T. T., Wong, K. Y., Zhang, X. M., and Xu, F. (2013). “A vision-based system for dynamic displacement measurement of long-span bridges: algorithm and verification.” *Smart Structures and Systems*, 12(3\_4), 363–379.
- Yoneyama, S., Kitagawa, A., Iwata, S., Tani, K., and Kikuta, H. (2007). “Bridge Deflection Measurement Using Digital Image Correlation.” *Experimental Techniques*, 31(1), 34–40.

- Yoneyama, S., and Ueda, H. (2012). “Bridge Deflection Measurement Using Digital Image Correlation with Camera Movement Correction.” *MATERIALS TRANSACTIONS*, 53(2), 285–290.
- Yoon, H., Elanwar, H., Choi, H., Golparvar-Fard, M., and Spencer, B. F. (2016). “Target-free approach for vision-based structural system identification using consumer-grade cameras: Target-Free Vision-based Structural System Identification.” *Structural Control and Health Monitoring*, 23(12), 1405–1416.
- Yoon, H., Shin, J., and Spencer, B. F. (2018). “Structural Displacement Measurement Using an Unmanned Aerial System: Structural displacement measurement using an UAS.” *Computer-Aided Civil and Infrastructure Engineering*, 33(3), 183–192.
- Zaurin, R., Khuc, T., and Catbas, F. N. (2016). “Hybrid Sensor-Camera Monitoring for Damage Detection: Case Study of a Real Bridge.” *Journal of Bridge Engineering*, 21(6), 05016002.

# Appendix A

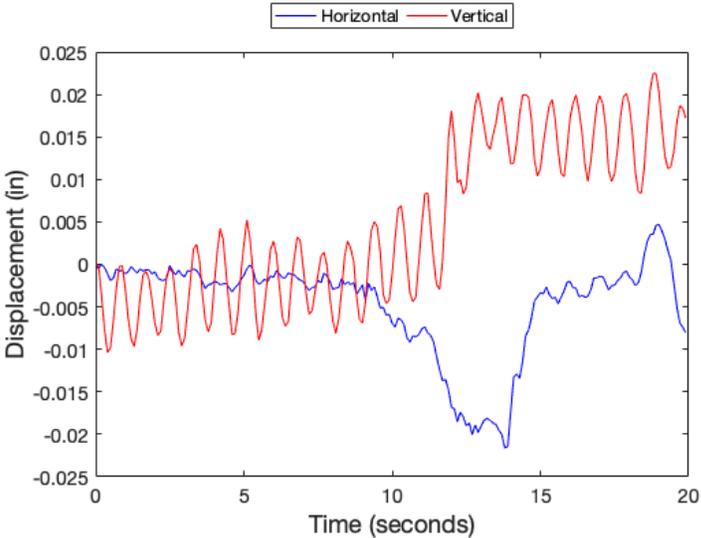


Figure A1. Traffic signal structure test: OFM displacement for test #1.

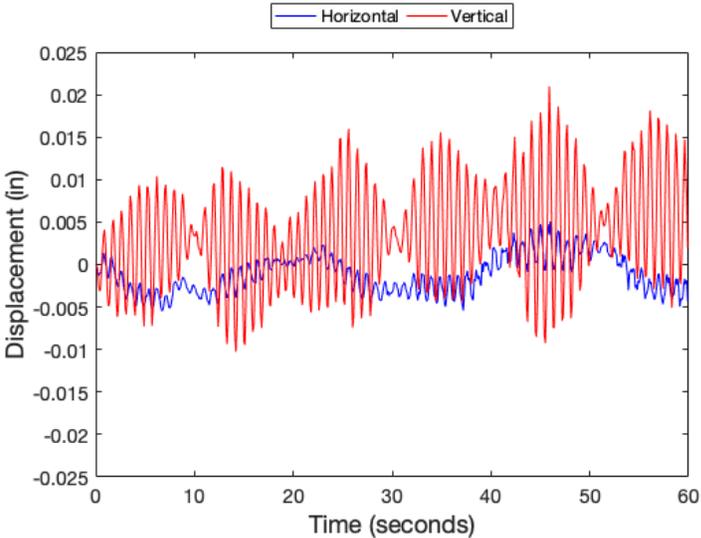


Figure A2. Traffic signal structure test: OFM displacement for test #2.

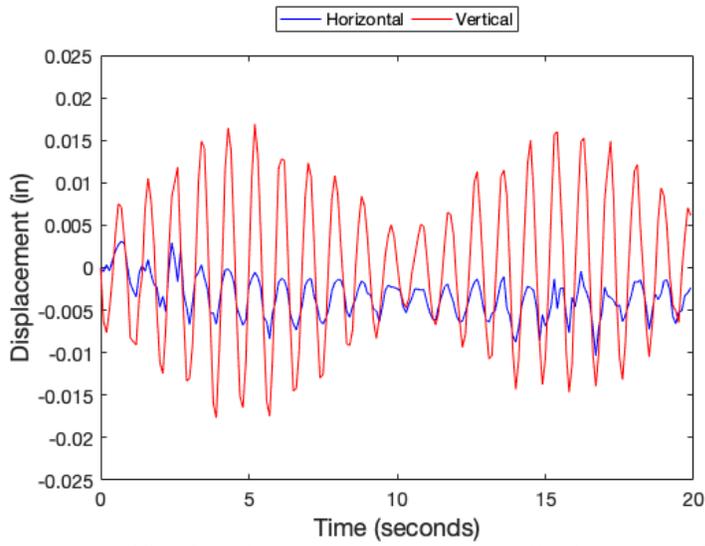


Figure A3. Traffic signal structure test: OFM displacement for test #3.

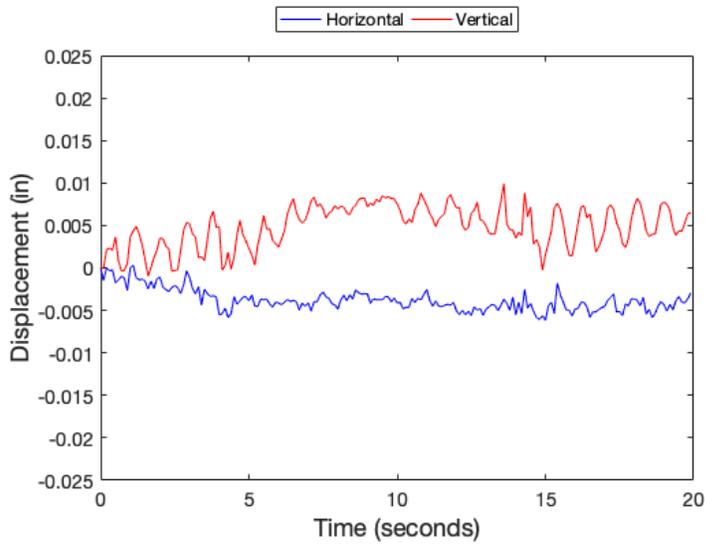


Figure A4. Traffic signal structure test: OFM displacement for test #4.

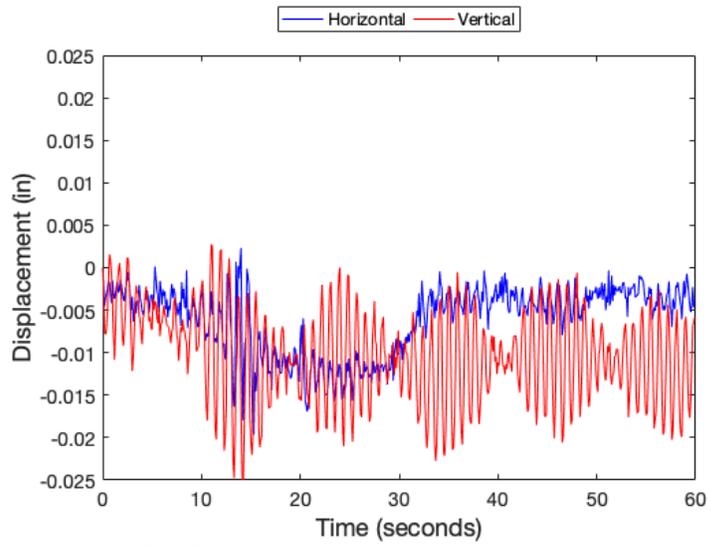


Figure A5. Traffic signal structure test: OFM displacement for test #5.

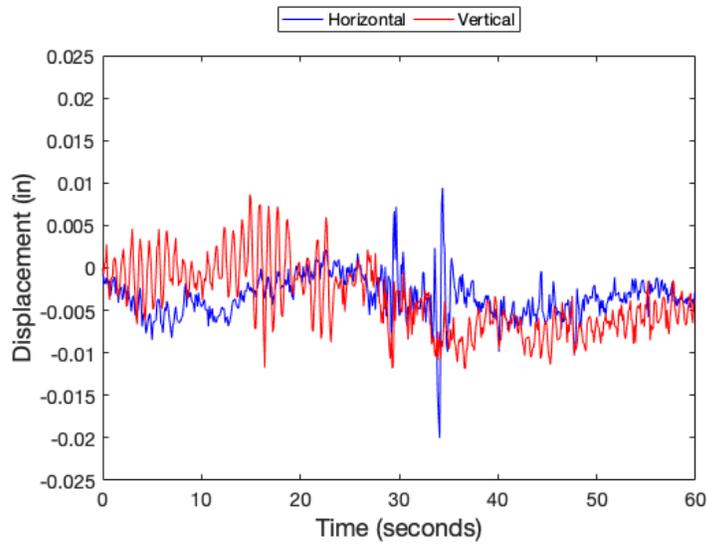


Figure A6. Traffic signal structure test: OFM displacement for test #6.