
Electronic Theses and Dissertations, 2004-2019

2016

Computer Vision Based Structural Identification Framework for Bridge Health Monitoring

Tung Khuc
University of Central Florida

 Part of the [Civil Engineering Commons](#)
Find similar works at: <https://stars.library.ucf.edu/etd>
University of Central Florida Libraries <http://library.ucf.edu>

This Doctoral Dissertation (Open Access) is brought to you for free and open access by STARS. It has been accepted for inclusion in Electronic Theses and Dissertations, 2004-2019 by an authorized administrator of STARS. For more information, please contact STARS@ucf.edu.

STARS Citation

Khuc, Tung, "Computer Vision Based Structural Identification Framework for Bridge Health Monitoring" (2016). *Electronic Theses and Dissertations, 2004-2019*. 4876.
<https://stars.library.ucf.edu/etd/4876>

COMPUTER VISION BASED STRUCTURAL IDENTIFICATION
FRAMEWORK FOR BRIDGE HEALTH MONITORING

by

TUNG KHUC

B.S. National University of Civil Engineering, Hanoi, Vietnam, 1999

M.S. University of Florida, Gainesville, FL, US, 2004

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the Department of Civil, Environmental, and Construction Engineering
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Spring Term
2016

Major Professor: F. Necati Çatbaş

© 2016 Tung Khuc

ABSTRACT

The objective of this dissertation is to develop a comprehensive Structural Identification (St-Id) framework with damage for bridge type structures by using cameras and computer vision technologies. The traditional St-Id frameworks rely on using conventional sensors. In this study, the collected input and output data employed in the St-Id system are acquired by series of vision-based measurements. The following novelties are proposed, developed and demonstrated in this project: a) vehicle load (input) modeling using computer vision, b) bridge response (output) using full non-contact approach using video/image processing, c) image-based structural identification using input-output measurements and new damage indicators. The input (loading) data due vehicles such as vehicle weights and vehicle locations on the bridges, are estimated by employing computer vision algorithms (detection, classification, and localization of objects) based on the video images of vehicles. Meanwhile, the output data as structural displacements are also obtained by defining and tracking image key-points of measurement locations. Subsequently, the input and output data sets are analyzed to construct novel types of damage indicators, named Unit Influence Surface (UIS). Finally, the new damage detection and localization framework is introduced that does not require a network of sensors, but much less number of sensors.

The main research significance is the first time development of algorithms that transform the measured video images into a form that is highly damage-sensitive/change-sensitive for bridge assessment within the context of Structural Identification with input and output characterization. The study exploits the unique attributes of computer vision systems, where the signal is continuous in space. This requires new adaptations and transformations that can handle computer vision data/signals for structural engineering applications. This research will significantly advance

current sensor-based structural health monitoring with computer-vision techniques, leading to practical applications for damage detection of complex structures with a novel approach. By using computer vision algorithms and cameras as special sensors for structural health monitoring, this study proposes an advance approach in bridge monitoring through which certain type of data that could not be collected by conventional sensors such as vehicle loads and location, can be obtained practically and accurately.

*To my Parents
and my Wife*

ACKNOWLEDGMENTS

First, I would like to express my thanks and my deepest gratitude to Dr. F. Necati Çatbaş; for his direction as well as his friendship throughout this dissertation. He has made this whole study possible. In addition, he taught me with experience and knowledge in research that assists my academia career in the future.

I would like to thank to the members of my dissertation committee, Dr. Ricardo Zaurin, Dr. Kevin R. Mackie, Dr. Mubarak Shah, and Dr. Amr Oloufa for their time and feedback. I am especially grateful to Dr. Ricardo Zaurin for his help and friendship during this study. I would also like to thank to Mr. Juan Cruz for his help in the UCF Structural Lab.

I would like to thank to Mehrdad Aghagholizadeh, Luca Letizia, Masoud Malekzadeh, AJ Modesto, Burak Gokce, and many other members of our research group. I especially thank to Ozan Celik for his unconditional help and friendship for my experiment as well as my writing.

TABLE OF CONTENTS

LIST OF FIGURES	x
LIST OF TABLES.....	xv
CHAPTER ONE: INTRODUCTION.....	1
Vision based Approach in Structural Health Monitoring Practices.....	1
Background of Bridge System	3
Research Goal and Objectives	4
Organization of Dissertation.....	7
CHAPTER TWO: COMPLETELY CONTACTLESS DISPLACEMENT MEASUREMENT OF STRUCTURES USING COMPUTER VISION BASED APPROACH.....	8
Introduction.....	8
Motivation and Objectives.....	11
Theoretical Background.....	13
Vision Acquisition System	14
New Virtual Markers - Image Key-points	15
Matching Key-points Between Consecutive Images	20
False matches and outlier discard algorithm using planar geometric transformation	24
Converting pixel based displacement to the engineering unit (e.g. inch, mm).....	26
Laboratory Verification	30
Experiment design	31
Laboratory results and discussion.....	32
Field verification on an elevated guideway for APM systems	37
Considerations for changing ambient illumination.....	38
Obtaining Structural Displacements and Identifying Dynamic Frequencies.....	39
Field implementation on a Stadium Structure	42
Stadium Displacements: Raw Data and Verification.....	44
Identification of the Modal Frequencies of the Structure	49
Accuracy of the Proposed Method.....	50
Conclusions.....	53

CHAPTER THREE: DETECTION AND LOCALIZATION OF VEHICLES ON A BRIDGE USING COMPUTER VISION APPROACH.....	55
Introduction.....	55
Motivation and Objectives.....	64
Theoretical Background.....	66
Imaging Features: Histograms of Oriented Gradients (HOG) Descriptors	67
Detection of Vehicle Types based on AdaBoost Technique and Cascade Classifier with HOG Features	69
Localizing Detected Vehicles based on Geometry Transform Approach	75
Laboratory Verification	80
Experiment Design: UCF 4-Span Bridge, Camera, and Vehicle Classes.....	80
Training Detectors using AdaBoost and Cascade Classifiers.....	82
Class Vehicle Detection Results, and False Detection	85
Vehicle Localization Results	88
Conclusions.....	92
CHAPTER FOUR: OBTAINING UNIT INFLUENCE SURFACES FROM VISION BASED MEASUREMENTS: A NEW STRUCTURAL DAMAGE INDICATOR.....	94
Introduction.....	94
Motivation and Objectives.....	99
Theoretical Background.....	102
Theory of Unit Influence Surface (UIS).....	102
Construction of Unit Influence Surface (UIS) from Direct Measurement	103
Surface Fitting for Discrete Data	107
Laboratory Verification	115
Experiment Setup on UCF 4-Span Bridge.....	115
Unit Influence Surfaces Results and Discussion	116
Conclusions.....	125
CHAPTER FIVE: DETECTION AND LOCALIZATION OF DAMAGE USING UIS OBTAINED WITH LIMITED NUMBER OF SENSORS	127
Introduction.....	127
Motivation and Objectives.....	131

Theoretical Background.....	133
Classical Approach: Damage Identification based on a Network of Measurement Positions	133
Detecting and Localizing Bridge Damage Simultaneously based on UIS's Obtained from Limited Measurement Locations	135
Laboratory Verification	137
Simulation of Damage Scenarios on UCF 4-Span Bridge.....	137
Damage Detection and Localization based on a Network of Measurement Locations ..	141
Detecting and Localizing Damage in Bridges Based on Limited Measurement Locations	153
Conclusions.....	164
CHAPTER SIX: CONCLUSIONS.....	166
Summary	166
Fully non-contact displacement measurement by means of image key-points:	166
Estimation of traffic vehicle weights and locations based on computer vision algorithms:	168
A new damage indicator for bridge St-Id system: Unit Influence Surface:.....	169
Bridge damage detection and localization based on limited measurement locations (sensors):.....	170
Contributions.....	171
Recommendations for Future Research	172
LIST OF REFERENCES.....	174

LIST OF FIGURES

Figure 1: Schema of the vision-based bridge St-Id system.....	5
Figure 2: Overview of the non-target vision-based displacement measurement method.....	14
Figure 3: Key-points (red dots) as corner points of a checkboard.....	18
Figure 4: Key-point identification using the local extrema detection algorithm – Modified from a figure in (Lowe, 2004).....	19
Figure 5: Detected key-points (red stars) on the bottom flange of the I-section steel girder at an elevation of more than 10 meters.....	20
Figure 6: (a) Key-points on image 1, (b) Key-points on image 2, (c) The matching result	23
Figure 7. (a) The matches without the outlier discarding algorithm, (b) Final matching result after utilizing the outlier discarding algorithm.....	25
Figure 8. Displacements of a measured location by matching key-points of its images along the video clip.....	26
Figure 9. Camera Theory	27
Figure 10. The relationship between the conversion ratio R vs. the camera distance of Z obtained for a Canon VIXIA HF R42 camcorder.....	30
Figure 11. The 4-span bridge at the UCF Structural Laboratory	31
Figure 12. Measured location and experimental setup	32
Figure 13. Comparison of displacement results by using the LVDT and the proposed method ..	33
Figure 14. Comparison of identified natural frequencies using the proposed method and the accelerometer data	36
Figure 15. (a) Trains running on the viaduct, (b) Runway under the viaduct.....	37

Figure 16. Measurement location and camera setup.....	38
Figure 17. Highly changing illumination among image frames due to shade of the trains	39
Figure 18. (a) Automatically matching the strongest key-points for low change of illumination region, (b) The best matches after utilizing the outlier discarding algorithm.....	39
Figure 19. Raw displacement data of a measurement location on obtained using the proposed method.....	40
Figure 20. Dynamic responses of the main girder extracted by the proposed method and an accelerometer (AS1-ACC25).....	41
Figure 21. Comparison of natural frequencies of the girder identified by the proposed vision-based method and accelerometers.....	42
Figure 22. The Football Stadium Monitored During Games	44
Figure 23. Experimental setup on the supported beam.....	45
Figure 24. Camera placement on a tripod.....	45
Figure 25. (a) Test 2 displacement datasets, (b) An zoom-in window from Test 2.....	47
Figure 26. (a) Test 3 displacement datasets, (b) Two zoom-in windows from Test 3.....	48
Figure 27. Comparison of natural frequencies of the supported beam identified by the proposed vision based method and processed acceleration data	50
Figure 28: Experimental setup for the method accuracy evaluation.....	51
Figure 29. Distributions of error data obtained from several accuracy evaluation tests.....	52
Figure 30. The relationship between the accuracy of the proposed measurement vs. the camera distance of Z obtained for Canon VIXIA HF R42 camcorder.....	53
Figure 31. Pre-weighted trucks deployed in a common bridge monitoring study	56

Figure 32. Protocol for detection and localization of traffic vehicles passing over a bridge based on computer vision approach	66
Figure 33. Protocol for determining HOG features; example for a given 128 x 128 pixels image window with cell size of 8 x 8 pixels and block size of 4 x 4 cells	68
Figure 34. HOG feature extraction represented as imaging fashion for example (a) a bike, (b) a sedan, and (c) a bottle	69
Figure 35. Schematic for a cascade of classifiers. Each level is a strong classifier that boosted from many weak classifiers	73
Figure 36. The training process for obtaining a detector	74
Figure 37. The detection scheme using a trained detector.....	75
Figure 38. Two detected vehicles are located on the 2-D image coordinates by bounding boxes; for example, the bottom-right corners of boxes are obtained in the pixel-unit.....	76
Figure 39. Geometry transformation between the 2-D image and the 2-D bridge deck coordinates	80
Figure 40. The UCF 4-Span Bridge and experiment setup.....	81
Figure 41. Vehicle classes and their specifications	82
Figure 42. Examples of positive and negative images for training classes (1,2,3) of vehicles	84
Figure 43. True positive of class vehicle detection.....	86
Figure 44. Examples of False Detection.....	87
Figure 45. Reference points on the 2-D bridge deck coordinate system (left) and on the 2-D image coordinate system (right)	89
Figure 46. An example of Unit Influence Surface.....	102

Figure 47. Protocol for extracting displacement Unit Influence Surface from direct measurements	104
Figure 48: A fitting surface constructed by using the bilinear interpolation	110
Figure 49: Spline interpolation vs. high order polynomial interpolation.....	111
Figure 50: Fitting surfaces of a given dataset obtained by two selected methods: a) Bilinear interpolation; b) Thin-plate spline interpolation	114
Figure 51: The twelve (12) measured locations on the UCF 4-Span Bridge girders.....	116
Figure 52: Experimental displacement data; a) The raw data and extracted static part at the location <i>L1</i> ; b) The static displacements at the locations from <i>L1</i> to <i>L6</i>	117
Figure 53: A combination data from the 10 tests acquired at the location <i>L2</i> for constructing displacement UIS at the <i>L2</i> (dUIS- <i>L2</i>); a) 3-D View; b) Plan View	118
Figure 54: The measured locations have geometrical symmetry. The UIS's at the locations in the same group should be similar	119
Figure 55: The displacement UIS's at the locations <i>L7</i> , <i>L8</i> , and <i>L9</i>	120
Figure 56: The displacement UIS's of the group <i>I</i> at the locations <i>L1</i> , <i>L6</i> , <i>L7</i> , and <i>L9</i>	121
Figure 57: The parameter values of the UIS's at the location <i>L1</i> for all <i>Sets</i> a) Volume of the UIS's; b) The maxima values of the UIS's	124
Figure 58: Establishing thresholds for the healthy structural scenario (<i>Baseline</i>).....	133
Figure 59: Establishing thresholds for the new UIS based damage identification	136
Figure 60: Damage simulation in the UCF 4-Span Bridge; a) Global damage by altering boundary condition at supports; b) Local damage by releasing composited connections	138
Figure 61: <i>Baseline</i> and <i>Damage</i> scenarios simulated in the UCF 4-Span Bridge.....	140
Figure 62: The measurement locations (<i>L1</i> to <i>L12</i>), and their groups (Group <i>I</i> , <i>II</i> , <i>III</i>)	141

Figure 63: Volume of UIS's for Baseline scenario: results and thresholds	144
Figure 64: Damage detection for <i>Damage 1</i> scenario	145
Figure 65: Damage detection for <i>Damage 2</i> scenario	146
Figure 66: Damage detection for <i>Damage 3</i> scenario	147
Figure 67: Damage detection for <i>Damage 4</i> scenario	148
Figure 68: Damage weight <i>DW</i> for <i>Damage 1</i> scenario	151
Figure 69: Damage weight <i>DW</i> for <i>Damage 2</i> scenario	151
Figure 70: Damage weight <i>DW</i> for <i>Damage 3</i> scenario	152
Figure 71: Damage weight <i>DW</i> for <i>Damage 4</i> scenario	152
Figure 72: The UIS matrix and its first derivative UIS-Y matrix.....	153
Figure 73: The mean, standard deviation and threshold matrices for the <i>Baseline</i> status	154
Figure 74: Damage recognition for an input data based on pre-developed thresholds of healthy condition	155
Figure 75: A section cut of damage identification process.....	156
Figure 76: Damage area for <i>Damage 1</i>	158
Figure 77: Damage area for <i>Damage 2</i>	158
Figure 78: Damage area for <i>Damage 3</i>	159
Figure 79: Damage area for <i>Damage 4</i>	159
Figure 80: Damage localization for <i>Damage 1</i>	162
Figure 81: Damage localization for <i>Damage 2</i>	162
Figure 82: Damage localization for <i>Damage 3</i>	163
Figure 83: Damage localization for <i>Damage 4</i>	163

LIST OF TABLES

Table 1. The conversion ratio R vs. the distance of Z	29
Table 2. The test specifications.....	32
Table 3. Correlation and error analysis for the proposed method.....	35
Table 4. Comparison of natural frequencies.....	42
Table 5. The test scenarios.....	46
Table 6. Correlation and error analysis.....	49
Table 7: Detection rates for each classes of vehicles.....	88
Table 8: Comparison of detection rates between cases of single vehicle and multiple vehicles on bridge deck.....	88
Table 9: Reference point coordinates: the world coordinate and the image coordinate systems (Column 2-5). The world coordinates calculated by the matrix T and errors (Column 6-9).....	91
Table 10: Error evaluation for the group I of the measured symmetrical positions.....	123
Table 11: The volume values of the UIS's for every Set	124
Table 12: The maximum values of the UIS's for every Set	124
Table 13: Percentage of damage $Cells$ for damage scenario	157

CHAPTER ONE: INTRODUCTION

Vision based Approach in Structural Health Monitoring Practices

Maintenance and safe operation of civil structures such as bridges, tunnels, stadiums, airports etc., require major investment and funds, especially in most of the developed countries where the civil engineering systems were built almost a hundred years ago. To help the owners of those structures manage their assets, assess the health of both old and new structures for maintenance and operation with limited funds, the need for efficient, cost-effective tools and technologies is very well established. Traditional maintenance and inspection of civil structures heavily rely on visual inspections, whose drawbacks have been sufficiently studied and pointed out (Phares, Washer, Rolander, Graybeal, & Moore, 2004). As an alternative practicing, Structural Health Monitoring (SHM) has been acknowledged as the most efficient approach for assessment of structures' health and performance, providing valuable information for better decision-making. Due to the increasing demand of structural assessment, SHM has been studied by numerous researchers worldwide especially over the last two decades.

In general, SHM utilizes advanced sensing technologies to measure the effects of input (naturally or artificially induced) and output (responses of structures) in order to track structural behavior towards more objective assessment of structures. Any change that is interpreted as an alteration of routine structural behavior can be a possible early onset of damage, raising a flag leading to a more in-depth investigation to produce information related to structural integrity. So far, many novel SHM methods, frameworks, and algorithms have been proposed and implemented for some certain cases but unfortunately there has not been a widespread adoption in practice. Several reasons to explain this observation are pointed out (F Necati Catbas & Kijewski-Correa,

2013). Hardware requirement of SHM systems is one of the challenges as many systems require complex field installation and maintenance for the sensors and the data acquisition (DAQ) systems. In addition, installation along with cost of these systems are generally quite expensive. Thus, possible use of non-contact measurement methods has received interest recently for inexpensive and/or practical field implementation purposes. It also motivates for paving a new route in SHM involving deployment of advanced image processing and computer vision techniques.

Vision based methods are acknowledged as a potential approach in SHM field due to emerging of related technologies (Webb, Vardanega, & Middleton, 2014). Undoubtedly, the most advantageous side of computer vision-based methods is non-contact implementation that accelerates inspection speed and reduces maintenance cost of structural assessments. Some challenges related to surface damage of structures are effectively solved by using image processing such as detection, and quantification of cracks and delamination of concrete (Jahanshahi & Masri, 2012; Kabir, Rivard, He, & Thivierge, 2009; Matsumoto, Mitani, & Catbas, 2015; T. Yamaguchi & Hashimoto, 2010; Yin, Wu, & Chen, 2014) and pavement (Oliveira & Correia, 2013; Zou, Cao, Li, Mao, & Wang, 2012). Measuring displacements of structures using vision based methods is also favored in the SHM community with numerous studies (Busca, Cigada, Mazzoleni, Tarabini, & Zappa, 2013b; Lee, Fukuda, Shinozuka, Yun, & Cho, 2007; Lee & Shinozuka, 2006; Santos, Costa, & Batista, 2012; Wahbeh, Caffrey, & Masri, 2003). Regarding structural identification systems for decision making, some researchers propose hybrid sensor-camera monitoring systems while utilizing cameras and computer vision techniques for obtaining traffic information inputs (F. Necati Catbas, Zaurin, Gul, & Gokce, 2012; Elgamal et al., 2003; Fraser, 2006; R Zaurin & Catbas, 2007; R. Zaurin & Catbas, 2010).

Background of Bridge System

Bridge network is always considered as one of the most essential components of civil infrastructure systems due to the vast amount of people and automobile that utilize it every day. For example, bridges are the key links in transportation grid helping passengers over natural obstacles such as rivers, valleys, or other roads. They have a direct influence on economy and human life. Well-operated bridges bring prosperity and convenience while damaged or collapsed bridges yields severe unwanted consequences such as terminated logistical flow and traffic jam.

As to structural engineering, bridges are special types of structures commonly working under difficult situations such as limited loading and harsh ambient conditions. Thus, it is seen that many bridges in the United States and their counterparts worldwide have been deteriorated and have lost some of their load carrying capacity over their life-cycles due to structural aging, lack of caring, and accidental effects. The 2013 assessment report issued by the American Society of Civil Engineers (ASCE) indicates that even though just below 11% of the nation's bridge are rated as structurally deficient, those bridges constitute one-third of total bridge decking area in the country. The report also states that \$20.5 billion is needed to invest annually to eliminate the deficient bridge backlog by 2028. It means that in case there is inexistence of effective methods for inspection and maintenance of the nation's bridges, the goal for eliminating the deficient bridges will be never accomplished since the annual budget is very limited.

It is acknowledged that a great deal of money can be saved by early diagnosis of bridge problems as well as fixing and retaining them in service rather than replacing the damaged components or rebuilding new structures. Although most bridges are assessed via periodic visual inspection, the drawbacks of this method have been well studied and pointed out (Aktan, Chase,

Inman, & Pines, 2001; Phares et al., 2004). For example, the main shortcomings are denoted as follows:

- ✓ *Incorrect Judgments of Visual Inspections:* So far, bridge inspection and maintenance have been visual tasks. Based on a schedule, inspectors visit a particular bridge followed by an inspection with bare eyes, and then subjective evaluation. Hence, the inspection results are biased depending on the personal opinions related to the experience of various inspectors. In addition, the challenge of access to numerous structural elements may cause inspectors to check possible deficiencies from distance, which results in incorrect identification of an initial damage. As a confirmation, a report issued by Federal Highway Administration's NDE Center reveals that 56% of the short-to-medium span bridges graded by visual inspection as being in average condition are incorrect (Turner Fairbanks Highway Research Center 2005).
- ✓ *Lack of Global Damage Evaluation:* The damages identified by bridge inspectors such as cracks, rusty members, leaking of members, etc. are considered as local damage. Even if these conclusions are correct, evaluating the effects of these damages to the global health of structures is still a challenge (Aktan et al., 2001).

Research Goal and Objectives

Motivated by the background of bridge system and also inspired by the promising applications of vision-based approach in SHM, the research goal is to propose a novel Structural Identification (St-Id) framework for assessment of bridge structures by deploying series of non-contact measurements based on advanced computer vision techniques. The collected vision based data is analyzed to form new types of damage indicators. Finally, the new damage indicators are

utilized for developing a damage identification algorithm (e.g. damage detection and damage localization). This research advances current sensor-based SHM for bridges by using camera-based implementation, enabling practical applications for broad adoption in Bridge Health Monitoring (BHM). The schema of research is illustrated as in Figure 1, and the detail objectives are listed as follows.

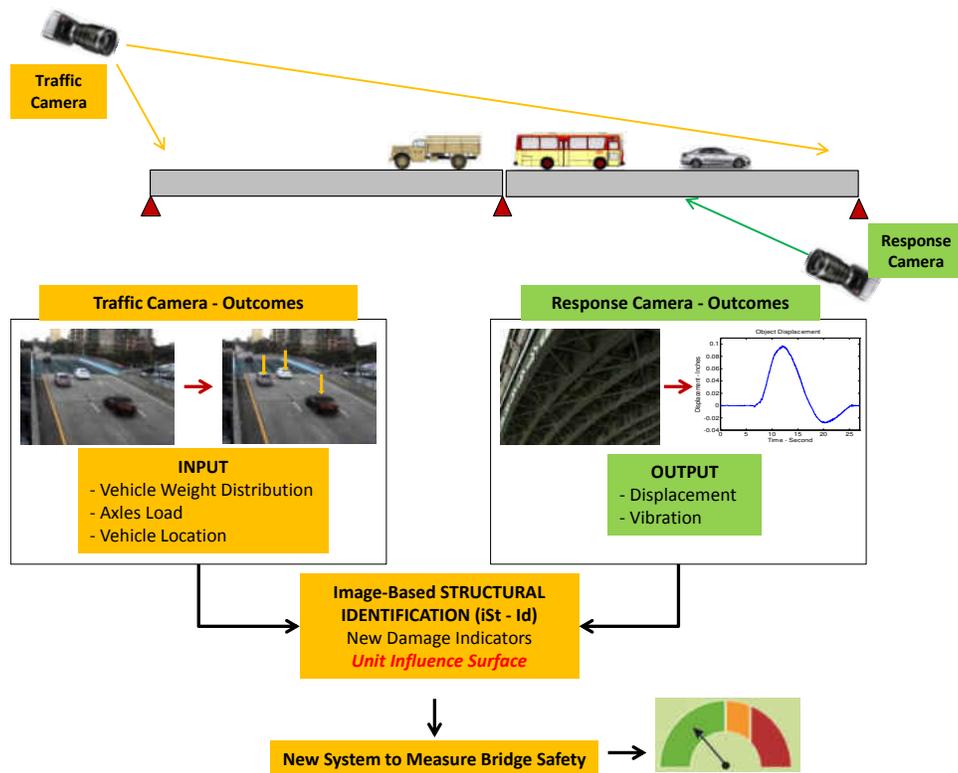


Figure 1: Schema of the vision-based bridge St-Id system

- ✓ *Objective 1 – A new fully non-contact displacement measurement for bridge structure by means of image key-points:* Selection of structural response type to be measured and method for acquisition is always the first step of any SHM implementation. Since displacement is a sort of powerful structural response that can be acquired wirelessly, introducing a fully contactless displacement measurement method is the first objective of this study. Besides, the fully contactless displacement measurement proposed for

this particular St-Id system can be used for different applications in civil engineering on a vast variety of structures.

- ✓ *Objective 2 – Developing a framework for estimating traffic vehicle weights and vehicle locations by employing object detection and localization algorithms on traffic surveillance images:* In general, Bridge Health Monitoring studies are implemented with the lack of vehicle information. It is due to the fact that the traffic vehicle data has not been comprehensively obtained yet (e.g. multiple vehicles, random route of vehicles etc.). In addition to that, the test-truck deployment requires permission for bridge closure. Thus, a framework for estimating traffic vehicle information such as weights and locations is proposed for complimenting the structural response data (i.e. structural displacements obtained as context in *Objective 1*).
- ✓ *Objective 3 – Construction of a new damage indicator for bridge health monitoring: Unit Influence Surface:* Since both the input data (traffic vehicle information) and the output data (structural displacements) can be obtained, a new damage indicator named the displacement Unit Influence Surface (UIS) is proposed. Unlike current damage indicators that solely demonstrate structural behaviors, the introduced damage indicator consists of not only structural responses but also geometric components of responses. It enables the capability of damage localization by using a single UIS. In addition, the UIS can be obtained by a non-contact method without conventional sensors. This is due to the fact that the data for UIS construction is acquired without any contact by cameras and vision techniques (ref. *Objective 1 & 2*).
- ✓ *Objective 4 – Bridge damage detection and localization based on limited measurement locations (sensors) by employing the new damage indicator UIS:* Although damages

can be directly detected by establishing a sort of binary classifiers from damage indicators (features), their locations are estimated based on corresponding sensor positions. Hence, a conventional damage identification requires a dense array of sensors. The proposed damage identification method has employed the geometric components of the new damage indicator UIS for localizing damage. Therefore, the framework can do both detection and localization without the need for multiple measurement locations or a sensor array. Since the new proposed framework does not require a network of sensors, it enables the possibility of a low-cost and convenient SHM implementation.

Organization of Dissertation

This dissertation is presented in the form of journal template. Except *Chapter 1* which is given as the introduction of the dissertation, four (4) previously mentioned *Objectives* are broken down into corresponding *Chapters 2, 3, 4, and 5* as independent journal papers. Each *Chapter* is introduced with the same template comprising of *Introduction, Chapter Objectives, Theoretical Background, Verifications, and Chapter Conclusions* sections. Finally, *Chapter 6* summarizes the theoretical highlights as well as the results of every previous *Chapters (2, 3, 4, and 5)*. The novelties and distributions of those proposals are also listed and pinpointed in *Chapter 6*. Furthermore, all disadvantages and challenges of the introduced frameworks are brainstormed and discussed, especially for further real-life implementation.

CHAPTER TWO: COMPLETELY CONTACTLESS DISPLACEMENT MEASUREMENT OF STRUCTURES USING COMPUTER VISION BASED APPROACH

Introduction

Reliably obtaining structural responses and tracking them for decision-making purposes is the first critical step for SHM. A change in dynamic and/or static response trend of a structure would be an indicator of damage occurring on the structure or some other structural issues that need to be evaluated. Most fundamental and common responses employed in SHM are acceleration, strain, tilt, displacement since these can clearly reflect both local and global behaviors of an existing structure under various loading conditions. Moreover, out of these common response types, displacement is arguably the most important one as the most developed performance-based design is direct displacement-based design where performance is related to acceptable damage and damage to displacement. As such displacement can be directly used for safety and serviceability limit state estimation despite displacement poses a particular measurement challenge due to reference requirement. Motivated by those reasons, this Chapter is aimed to develop a completely contactless, cost-effective and practical displacement measuring method for real-life structures where displacement monitoring might not be easy or possible.

Traditionally, displacement sensors such as linear variable differential transformers (LVDTs), slide wire potentiometer (SWPs) or dial gauges have been utilized to collect displacement responses. These classical sensors are quite convenient to use in laboratories; however, are not practical to deploy on a real-life structure due to several reasons such as the need for stationary platforms near measurement points to mount sensors, and limitation of the sensor range. There are other proposed approaches to tackle those drawbacks of classical sensors such as

Global Position System (GPS), Interferometric Radar, Laser Doppler Vibrometer (LDV), and Scanning Laser Vibrometer. Although radar and laser based methods provide high precision, they require very high cost equipment. Currently, the GPS system costs are coming down, however the limitations due to GPS accuracy as well as possible sampling rates remain as issues to be solved. Those limitations make the GPS system be commonly suitable for specific applications such as cable-bridge monitoring studies due to those structures have larger displacement range and low natural frequencies (Im, Hurlbaas, & Kang, 2011). To address all above limitations, vision-based monitoring has been explored due to its practical deployment and cost-effectiveness. Regarding vision-based displacement and vibration measurement, several studies proposed algorithms for determining deflection and vibration from multi-points on a small beam by means of matching detected edges or markers between consecutive image frames collected by a digital camera (Cantatore, Cigada, Sala, & Zappa, 2009; Jurjo, Magluta, Roitman, & Gonçalves, 2010; Patsias & Staszewski, 2002; Poudel, Fu, & Ye, 2005; Rucka & Wilde, 2005; Shi, Xu, Wang, & Li, 2010; Sładek et al., 2013). Even though most of these studies could obtain both static deflection and dynamic vibration of a beam, the algorithms were limited for laboratory implementations. This is due to the fact that the studies were only practical for a small structure, hence the entire structure could be taken inside an image view.

Conducting SHM studies for real-life structures using computer vision-based techniques has been explored by some researchers due to practical nature of the measurements. In 2003, Jauregui et al. conducted a series of tests in New Mexico State to measure displacements of targets attached under main girders of several bridges by identifying their three-dimensional (3-D) locations (Jauregui, White, Woodward, & Leitch, 2003). By obtaining images of targets at different viewpoints, the authors successfully determined 3-D locations of targets by utilizing the

principle of triangulation algorithm (a basic computer vision technique). Another research to determine displacements of a real structure was implemented in 2006 at a steel bridge in Korea (Lee & Shinozuka, 2006). In this study, the authors developed a practical vision system that could obtain displacement data at real-time speed. Using a special target containing four black dots on white background, these dots could be detected in terms of color filtering for tracking their motions in time domain. Moreover, the pre-defined distances among the dots had be utilized for converting from pixel unit to engineering unit (millimeter). That approach was later improved to obtain displacements from multiple locations of structures by synchronizing numerous vision systems with a wireless network (Lee et al., 2007).

Application of normalized cross correlation (NCC) imaging algorithm has become quite popular for vision-based displacement measurement methods. By calculating correlation of ROIs (region of interest) of two target images, movements of the ROIs between consecutive imaging frames were determined even at sub-pixel accuracy. A comprehensive study using this NCC approach for vibration monitoring was carried out (S.-W. Kim & Kim, 2013). In that study, several powerful image processing techniques were employed such as digital image correlation (DIC), image transform function (ITF), and sub-pixel analysis to make their method robust and precise. The study had been conducted for assessment of cables on a suspension bridge including natural frequency estimation and cable-tension verification. Some other researchers deployed the NCC based displacement monitoring systems for assessment of bridge sub-structures (Sung, Miyasaka, Lin, Wang, & Wang, 2012) as well as for finite element model updating (S.-W. Kim, Lee, Kim, & Kim, 2013). That approach was also successfully conducted on laboratory experiments and on several other bridges in Korea and Hong Kong (S.-W. Kim et al., 2013; Ye et al., 2013). Lately, some targetless practices have also been implemented by utilizing the NCC imaging algorithm on

natural textures of structure images. However, only pixel-based vibration for identifying structural characteristics was interested during these experiments (S. W. Kim, Jeon, Kim, & Park, 2013). Additionally, inconsistent results were observed due to the effects of low contrast on structure textures (Busca, Cigada, Mazzoleni, Tarabini, & Zappa, 2013a).

Motivation and Objectives

Although vision-based SHM systems for structural displacement and vibration monitoring have been proposed over a decade, the main requirement of these systems has not really changed. A general framework for conducting a vision-based displacement measurement includes (i) Capturing video clips of targets attached on monitored locations using an extra-tele lens camera, (ii) Identifying dominant regions on targets by means of image processing, (iii) Determining image features of these dominant regions to match them between consecutive frames, and (iv) Calculating pixel-based displacements; and then converting them to the engineering unit (millimeter) by using standards on the targets (e.g. pre-defined shape dimensions). Following these steps, a target (e.g. a plate consisting of figures such as circles or rectangles with known dimensions) plays ultimately important roles for image processing such as being a dominant marker of measurement location, and being a conversion standard for transferring displacements from pixel unit to engineering unit (e.g. inch, millimeter). Target attachment on real life structures such as bridges, high rise buildings, towers etc. may be cumbersome in many cases due to the limited access to target locations as well as the effort needed for target attachment using specialized equipment such as a snooper truck and high scaffold systems. Requirements to utilize targets make such vision-based displacement measurement not a fully non-contact method, limiting the real life implementations so far. Besides this limitation, most of previous vision-based displacement measurement methods lack of a

comprehensive study for evaluating accuracy of the proposals. Since users do not know the accuracy of these methods (e.g. what is the minimum displacement the method can reliably be obtained?), it is difficult for them to agree upon using a new measurement method/approach in their SHM studies.

The goal of this Chapter is to further improve displacement monitoring by developing a non-target vision-based method that will address the limitation of target attachment. To discard the physical targets in general vision-based displacement monitoring framework, three objectives are proposed as follows.

- ✓ Exploration of a new type of virtual marker on measurement locations called imaging key-points that can replace conventional physical targets
- ✓ Development of a conversion method based on the camera calibration technique to transfer pixel-based displacements to engineering-based (millimeter) displacements since physical targets no longer exist.
- ✓ Proposing approaches to handle challenges such as low contrast, changing illumination and outliers in matching key-points

By achieving fully non-contact monitoring, implementing the proposed method will be more practical, especially for real-life structures. Without using the target attachment in a vision-based displacement monitoring, most of the field works and requirements such as installing targets, sensors, and DAQ systems, as well as wiring cables are not needed anymore. That improvement enables not only a cost-effective measurement method but also a possibility to obtain structural responses from difficult access locations. With the implementation advantages plus the generic response that can be obtained, the method gives an opportunity for developing a more comprehensive and practical SHM framework.

The proposed framework is validated on a 4-Span Bridge model at the University of Central Florida (UCF) Structures Laboratory. Subsequently, field verifications are conducted on an elevated guideway structure and on a real-life stadium during foot-ball games. The obtained results have successfully been verified using those conventional sensors such as LVDTs and accelerometers, which are presented on subsequent sections.

Theoretical Background

The implementation of the non-target vision-based displacement and vibration monitoring is summarized in six steps (Figure 2). First, an image sequence of a particular measurement position is acquired using a low-cost camera. These images are processed to extract the key-points, which are then matched between consecutive frames to determine pixel based displacements. The pixel based displacements are then converted to engineering units (e.g. inch, millimeter).

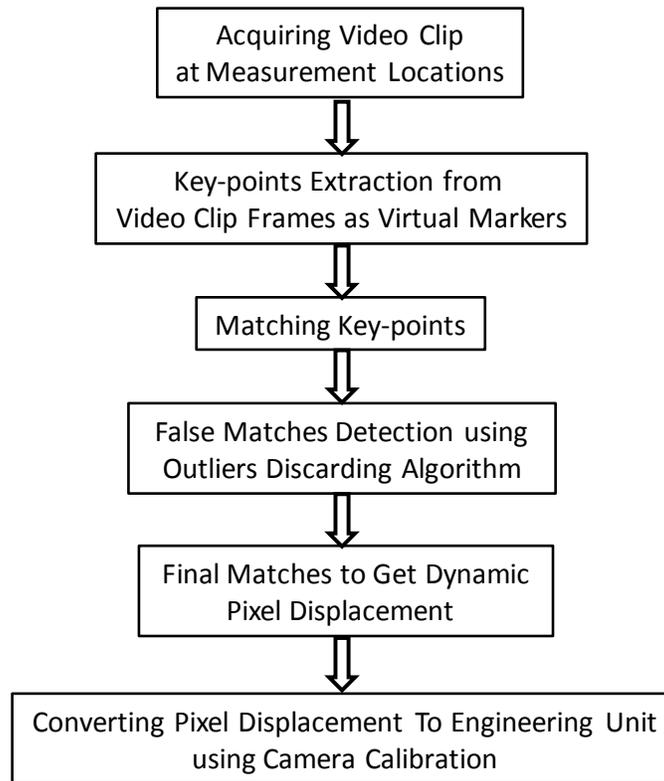


Figure 2: Overview of the non-target vision-based displacement measurement method

Vision Acquisition System

As an important goal of this study, a low-cost and practical vision acquisition system is designed so that data can be acquired very easily. Minimum requirements for this vision system include:

- ✓ A low-cost high density (HD) camcorder with a sample rate of 60 frames per second (e.g. Canon VIXIA HF R42 or similar ones)
- ✓ A laser distance measurer which can obtain both distance and angle between the measurement location and the camera
- ✓ Miscellaneous equipment such as a tripod, a checkerboard, etc.

New Virtual Markers - Image Key-points

Image matching is a fundamental aspect of many problems in computer vision including object or scene recognition, rebuilding 3D structure, stereo, and motion tracking (Lowe, 2004). To match different images of the same object, researchers commonly extract image features of the object that are invariant to such as image translation, rotation, scaling and illumination changing. The image matching technique utilizes a general procedure for vision-based displacement monitoring while the matched and tracked objects are measurement positions. Traditionally, previous studies for vision-based displacement measurement employ physical targets as a type of predefined image feature; consequently, target attachment is a requirement. A target including known-dimension circles or rectangles provides dominant markers (e.g. center and/or corner points of those geometrical shapes) for matching and tracking easily by means of basic image processing algorithms. In this paper, due to the shortcomings of using target attachment as mentioned in the previous sections, image key-points (a natural image feature) are used as virtual markers of measurement locations that replacing any physical targets.

In computer vision field, a key-points is defined as a special pixel that has dominant textures or characteristics comparing to its neighbors. Even though there are different types of key-points obtained by different computer vision algorithms, the robust key-points are interested due to their invariance, reliability and consistency. Once the robust key-points on monitoring positions are detected, motions of these structural locations will be determined in terms of the key-points movements, which can be tracked across consecutive image frames. In the pool of extracting robust key-points algorithms, following methods have been acknowledged as the most robust techniques including Harris Corner Point (Harris & Stephens, 1988), Scale Invariant Feature Transform (SIFT) (Lowe, 2004), Speed-up Robust Feature (SURF) (Bay, Ess, Tuytelaars, & Van Gool, 2008),

Binary Robust Invariant Scalable Key-points (BRISK) (Leutenegger, Chli, & Siegwart, 2011), and Fast Retina Key-points (FREAK) (Alahi, Ortiz, & Vanderghenst, 2012). In this study, the Harris Corner Point and the Scale Invariant Feature Transform (SIFT) methods are explained due to their basic and fundamental insights, which are then upgraded and expanded for the next generation of key-point related algorithms.

Harris Conner Points

One of the beginning algorithm for extracting a sort of key-points is introduced by Harris and Stephens (Harris & Stephens, 1988). In that study, the key-points are acquired by means of deploying the Hessian matrix, a square matrix of second-order partial derivatives of a multivariable function. Since an image can be represented as a discrete two-variable function (or so-called a matrix) of intensity values $I(x, y)$ with respect to x -row and y -column, the Hessian matrix of an image describes the local curvature of the function $I(x, y)$, which is shown in Equation 1.

$$H = w \otimes \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (1)$$

where \otimes is a mathematical operator called convolution applied to two matrices in this certain case. A Gaussian kernel w is a filtering window, which is commonly convolved with an image to discard noises. I_x^2 and I_y^2 are second-order derivatives of the image corresponding to x -coordinate and y -coordinate; while $I_x I_y$ is a mixed derivative of the image with respect to both x and y -directions. The first order derivatives I_x and I_y of the image can be found by using different derivative kernels k_x and k_y , with respect to x and y -coordinates respectively. Then, the second order derivatives I_x^2 , I_y^2 and $I_x I_y$ are determined by convolving the kernels k_x and k_y with the first order derivatives as follows.

$$k_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \text{ and } k_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (2)$$

Where

$$\begin{aligned} I_x &= k_x \otimes I(x, y) \\ I_y &= k_y \otimes I(x, y) \end{aligned} \quad (3)$$

And

$$\begin{aligned} I_x^2 &= k_x \otimes I_x \\ I_y^2 &= k_y \otimes I_y \\ I_x I_y &= k_y \otimes I_x \text{ or } = k_x \otimes I_y \end{aligned} \quad (4)$$

Due to the Gaussian filtering window w being a constant matrix, this part can be embedded inside the Hessian matrix, hence, Equation 1 can be rewritten as follows.

$$H = \begin{bmatrix} A & C \\ C & B \end{bmatrix} \quad (5)$$

where A, B, and C are matrices determined as

$$\begin{aligned} A &= w \otimes I_x^2 \\ B &= w \otimes I_y^2 \\ C &= w \otimes I_x I_y \end{aligned} \quad (6)$$

To detect the key-points of the image, Harris and Stephens proposed a response matrix F (Equation 7) by analyzing the eigenvalues of the Hessian matrix (Harris & Stephens, 1988). Then, once a threshold for F is chosen and utilized, regions of positive key-points can be identified. To obtain key-point locations, the local maxima algorithm is executed on these positive key-point regions at a sub-pixel level of accuracy. For example, the key-points obtained as corner points of a checkboard is shown in Figure 3 by implementing the Harris corner detection.

$$F = AB - C^2 - k(A + B)^2 \quad (7)$$

where k is an empirical factor, $k = 0.04-0.06$.

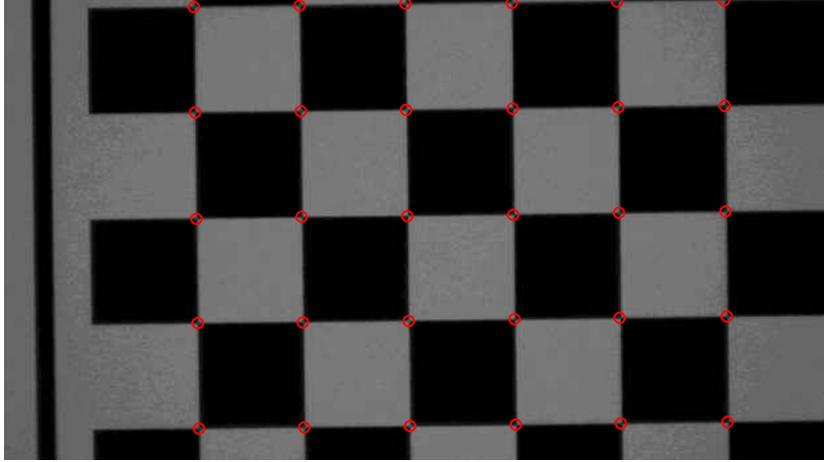


Figure 3: Key-points (red dots) as corner points of a checkboard

Scale Invariant Feature Transform (SIFT) Algorithm

The SIFT algorithm is a robust key-point detection technique that can extract the key-points that are highly invariant with respect to rotation, translation, scale, and changing of illumination condition. Following the SIFT method, an input image $I(x, y)$ is filtered by using the Gaussian kernel to discard noise that commonly dominate key-point candidates. Since it is impossible to find the most suitable Gaussian kernel, a scale-space of Gaussian functions corresponding to different standard deviation values of σ_i , namely $G(x, y, \sigma_i)$, is utilized to scan all potential candidates of key-points at every scale of filtering. The scale space of filtered images $L(x, y, \sigma_i)$ is derived following Equation 8.

$$L(x, y, \sigma_i) = G(x, y, \sigma_i) \otimes I(x, y) \quad (8)$$

Where

$$G(x, y, \sigma_i) = \frac{1}{2\pi\sigma_i^2} e^{-(x^2 + y^2)/2\sigma_i^2} \quad (9)$$

To efficiently detect key-point locations, differences of two adjacent filtered images (e.g. filtered images $L(x, y, \sigma_i)$ and $L(x, y, \sigma_{i+1})$) have been determined in which the local extrema detection algorithm is then applied. Those differences of filtered images $D(x, y, \sigma_i)$ can be calculated as follows.

$$D(x, y, \sigma_i) = L(x, y, \sigma_{i+1}) - L(x, y, \sigma_i) \quad (10)$$

Subsequently, a key-point can be detected at the location which has a local extrema value on the differences of filtered images $D(x, y, \sigma_i)$. That local extrema detection process is to compare the candidate intensity value (marked with **X**) to its (26) neighbors in 3 x 3 regions at the current and adjacent scales (marked with circles) as shown in Figure 4. Consequently, the key-points found from the previous step have been tested their robustness to reject the low contract and the poor location (e.g. along an edge) candidates. The detailed explanation as well as all related equations can be found in (Lowe, 2004).

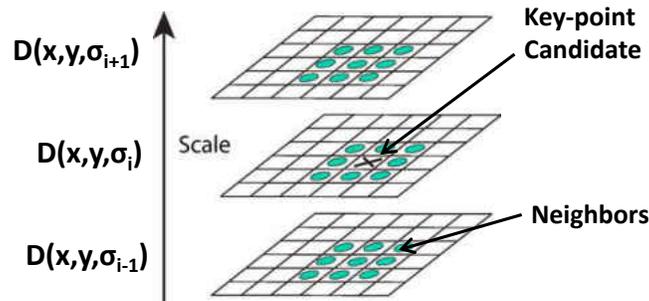


Figure 4: Key-point identification using the local extrema detection algorithm – Modified from a figure in (Lowe, 2004)

Figure 5 shows the detected key-points from an image of a measurement position located under the bottom flange of an I-section steel girder. The girder is from a football stadium and is

supported at an elevation of more than 10 meters. Although the coating paint layer makes the image low in contrast, more than sixty (60) key-points were detected around the bolt and along the weld on the member.



Figure 5: Detected key-points (red stars) on the bottom flange of the I-section steel girder at an elevation of more than 10 meters

Matching Key-points Between Consecutive Images

The key-points are obtained from a sequence of images following the process described in the previous section. To track the key-points over time, key-point matches between two consecutive images need to be determined by matching their descriptor vectors. Commonly, a descriptor vector of a key-point is extracted from image intensity values of a small patch around the key-point since its neighborhood tells more information than itself. A good descriptor describes the patch in a way that it is invariant with respect to the image changes (rotation, translation, and changing of illumination condition, etc.). As an intuitive example, fingerprint is a very good descriptor to identify a person. Although many algorithms have been proposed to obtain different types of descriptors, they can be categorized into two families: gradients and orientations based descriptors and binary based descriptors.

In the first group, researchers have computed the gradients of pixels in the patch which are invariant to light intensity. Then, these gradients have been concatenated into a histogram which is invariant to rotation to explore orientation of sub-cells on the patch. The descriptor has become a vector presenting all gradients and orientations of the patch sub-cells (Bay et al., 2008; Lowe, 2004). Recently, a new approach has received the attention of the computer vision community since it has encoded characteristics of the patch as a binary vector by sampling many logical connections ("0" and "1" values) among pixels on the patch (Alahi et al., 2012; Calonder, Lepetit, Strecha, & Fua, 2010; Leutenegger et al., 2011). Those logical links may be assigned randomly, or pre-defined patterns may be followed. Even though it is difficult to tell what the approach outperforms when compared to others, the binary based descriptors can be obtained at a very fast computation speed compared to gradient and orientation based descriptors.

Gradients and Orientations based Descriptors

Following the approach, a descriptor vector of a key-point is built up from the gradient magnitudes and orientations of the key-point neighbors, which can be calculated as following Equation 11 & 12. The details of development of descriptor vector from the gradient magnitude and orientation values are illustrated in (Lowe, 2004).

$$m(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^2 + (L(x, y + 1) - L(x, y - 1))^2} \quad (11)$$

And

$$\theta(x, y) = \tan^{-1} \left(\frac{L(x, y + 1) - L(x, y - 1)}{L(x + 1, y) - L(x - 1, y)} \right) \quad (12)$$

where $m(x, y)$ is the gradient magnitude, and $\theta(x, y)$ is the orientation of each neighboring pixel.

Once descriptor vectors are determined for every key-points, a key-point on a certain image (e.g. image k^{th} on a video sequence) will be matched to another key-point on the next image $(k+1)^{\text{th}}$ by identifying its nearest neighbor. The nearest neighbor is obtained by determining the minimum Euclidean distance between the key-point descriptor vector on the image k^{th} and all other key-point descriptor vectors from the image $(k+1)^{\text{th}}$.

Binary based Descriptors – FREAK algorithm

Proposed by Alahi et al., the FREAK algorithm is the most recent binary based descriptors that utilizes a series of pre-defined logical link patterns inspired by the distribution of receptive fields over the human retina (Alahi et al., 2012). The "0" or "1" value of each link in these pre-defined logical link patterns are simply assigned by comparing the intensity values of two pixels at the link ends following Equation 13.

$$L = \begin{cases} 1, & \text{if } I_2 - I_1 < 0 \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

Where L is a logical link value; I_1 and I_2 are the intensity values at the beginning and ending pixels of the link, respectively. Consequently, the FREAK descriptors (V) can be determined as a 512-dimension binary vector which demonstrates the local region around the key-point.

$$V = \{L^1, L^2, \dots, L^{512}\} \quad (14)$$

To match the key-points between two images, their descriptors are compared. As descriptors are represented by binary strings, their comparisons are carried out by means of measuring the Hamming distances (H) between two binary vectors (V^a and V^b). The shortest Hamming distance indicates the best match.

$$H = \sum (V^a \text{ XOR } V^b) \quad (15)$$

where XOR is a logical operation called “*exclusive or*” that outputs true (1 value) whenever both inputs differ and false (0 value) whenever these inputs turn out to be the same.

The matching of two key-point sets extracted from two images deploying the binary based descriptors is illustrated in Figure 6. Even though the number of key-points obtained from image 1 and 2 are 60 and 37 respectively, the number of matches determined by the matching algorithm is 18. Again, similar to the results obtained by deploying the gradients and orientations based descriptors, the results shows some false matches that can even be recognized visually.

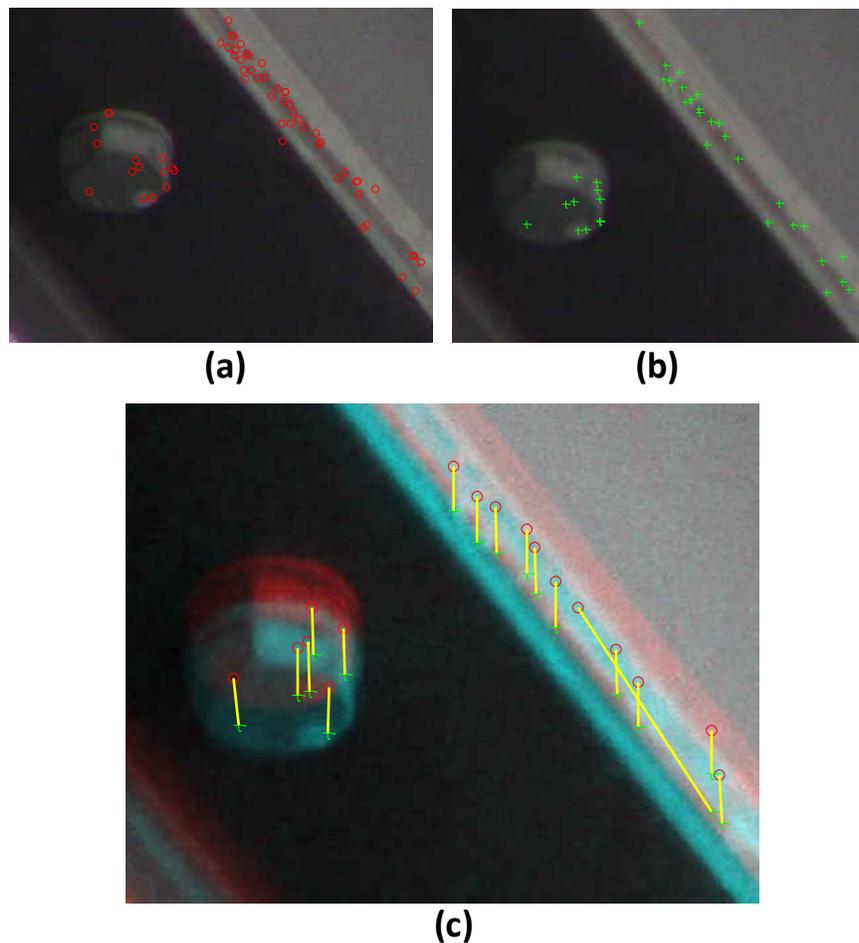


Figure 6: (a) Key-points on image 1, (b) Key-points on image 2, (c) The matching result

False matches and outlier discard algorithm using planar geometric transformation

Although using descriptors for matching key-points can be considered as the state of the art approaches, false-matching problem can be observed. The problem occurs more frequently when the tests are conducted under change of illumination or bad light conditions, which is a common issue when collecting video clips in the field from real structures. To discard the outliers in the pool of matches, a relationship function between two matched key-point sets is determined by using the Geometric Transformation technique. Let S_i and S_{i+1} be the two sets of matched key-points obtained from two consecutive images i^{th} and $(i+1)^{th}$, so

$$S_i = \{K_i^1, K_i^2, \dots, K_i^{n-1}, K_i^n\}$$

And (16)

$$S_{i+1} = \{K_{i+1}^1, K_{i+1}^2, \dots, K_{i+1}^{n-1}, K_{i+1}^n\}$$

Where n is the number of key-points in each matched set; and K is R^2 - coordinates (x, y) of key-points. A particular matched pair of key-points (K_i^j, K_{i+1}^j) is detected as the outliers if the distance between the projection of K_i^j , which is calculated by the relationship function, and K_{i+1}^j is higher than a pre-set threshold value, for instance, one pixel.

The relationship between S_i and S_{i+1} can be expressed in Equation 17 and Equation 18 as follows.

$$S_i = T S_{i+1} \tag{17}$$

Or

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{i+1} \\ y_{i+1} \\ 1 \end{bmatrix} \tag{18}$$

where T is the transformation matrix, or the relationship function that maps i^{th} image coordinates to $(i+1)^{th}$ image coordinates.

$$T = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \quad (19)$$

The T matrix is established by using a common approach named as the Least Squares Fit algorithm for an over-determined system. As a consequence, matches which are not compatible with the matrix of T are assigned as outliers and are discarded out of the matched key-point sets. The result of this procedure is illustrated in Figure 7.

The procedure for matching and then rejecting outliers will be repeated for all images along the video clip in time domain; and displacements of the measured locations in time history is illustrated in Figure 8.

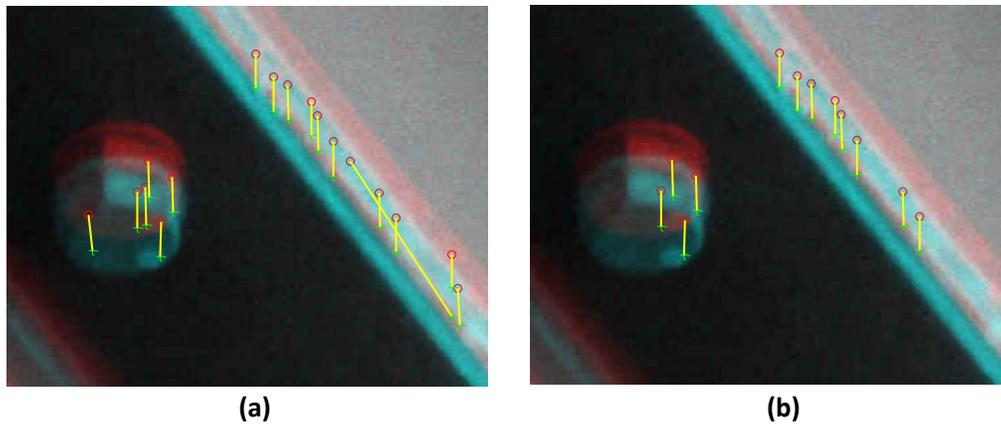


Figure 7. (a) The matches without the outlier discarding algorithm, (b) Final matching result after utilizing the outlier discarding algorithm

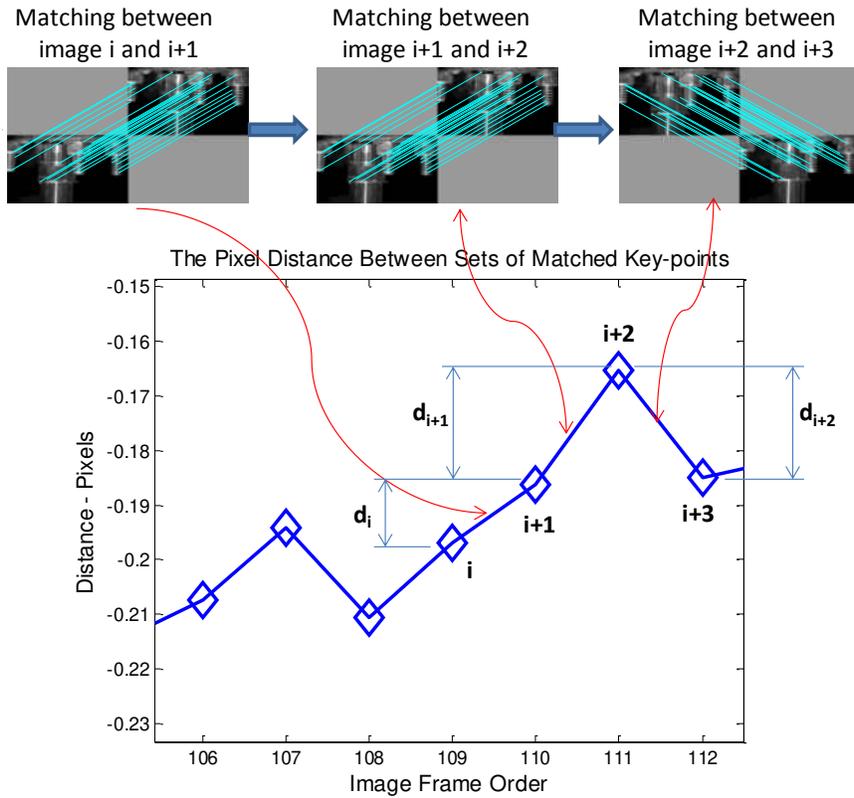


Figure 8. Displacements of a measured location by matching key-points of its images along the video clip.

Converting pixel based displacement to the engineering unit (e.g. inch, mm)

It is common to use the physical target not only as a marker but also as a sort of calibrating standard in vision-based methods. By knowing the distance between two pre-selected points on the target in the image coordinate and the world coordinate, a conversion ratio between pixels and engineering units can easily be determined from Equation 20.

$$R = \frac{d}{D} \left(\frac{\text{pixel}}{\text{mm}} \right) \quad (20)$$

Where R is the conversion ratio between the image coordinate and the world coordinate; d is the distance in the image coordinate (pixel); and D is the distance in the world coordinate (mm).

Even in case of no physical targets, the value of d can easily be determined by measuring an imaging object in the pixel unit; however, obtaining the D value may be impossible due to the real object being in a position that is out of reach for real-life structures. Here, an alternative method is proposed to tackle this problem by establishing a relationship between the conversion ratio R and the distance Z from the camera to measurement locations in terms of camera calibration. Following the camera theory as illustrated in Figure 9, the value of d can be calculated from Equation 21.

$$d = \frac{h}{p} \quad (\text{pixel}) \quad (21)$$

Where h is the object image dimension in the engineering unit (mm); p is the unit length of camera sensor (mm/pixel), which may be provided by the camera manufacturer. In addition, the triangle similarity theorem allows Equation 22 to be used.

$$\frac{h}{D} = \frac{f}{Z} \quad (\text{pixel}) \quad (22)$$

Where f is the focal length of camera; Z is the distance from the camera to the object (measurement location).

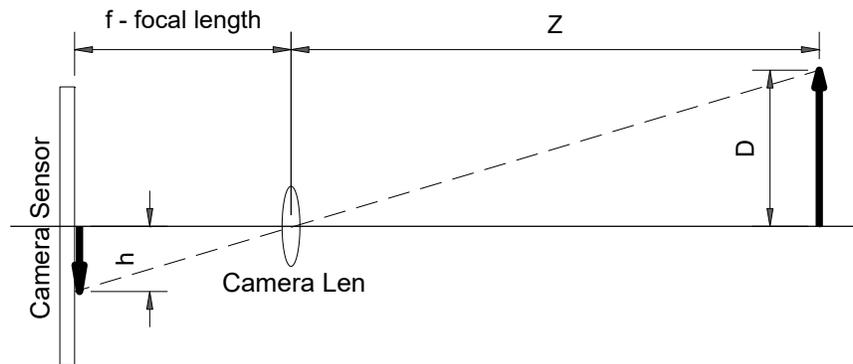


Figure 9. Camera Theory

Substituting Equation 21 and Equation 22 into Equation 20 and rearranging gives Equation 23, in which the conversion ratio R is demonstrated as an inversely proportional function to Z value.

$$R = \frac{f}{p * Z} \quad \left(\frac{pixel}{mm}\right) \quad (23)$$

Generally, the ratio $\frac{f}{p}$ can be calculated via the camera specifications provided by its manufacturer; however, this information can hardly be found for the majority of low-cost cameras. Due to this reason, the relationship between R and Z values is determined by using a camera calibration algorithm. Calibration of a camera is always one time implementation and it is not related to the monitoring framework. In this study, a Canon VIXIA HF R42 camcorder is calibrated using a checkerboard which possesses 7x10 square shapes of alternating black and white color. The dimensions of a unit square are 14.4x14.4mm. By keeping the same zoom factor of the camera after collecting video clips from the measurement locations (normally at the highest zoom factor), the checkerboard is captured at a series of pre-defined distances of Z_{cal} . The *conversion ratio* R_{cal} is calculated by Equation 24.

$$R_{cal} = \frac{d_{cal}}{D_{cal}} \quad \left(\frac{pixels}{mm}\right) \quad (24)$$

Where d_{cal} is the space between two adjacent corner points on the checkerboard measured in the image coordinate (pixel). This value can be obtained by utilizing a simple image processing on the checkerboard image such as using Harris corner detection method (Harris & Stephens, 1988). The value of D_{cal} is the dimension of the unit square on the checkerboard (14.4 mm). The results of R_{cal} values with respect to the variables of Z_{cal} are shown in Table 1.

Table 1. The conversion ratio R vs. the distance of Z

	Z_{cal} (m)	D_{cal} (World) (mm)	d_{cal} (Image) (pixel)	R_{cal} (pixel/mm)
Test 1	13.8	14.4	64.6	4.474
Test 2	12.9	14.4	68.9	4.775
Test 3	11.9	14.4	74.1	5.135
Test 4	11.0	14.4	79.6	5.516
Test 5	10.1	14.4	86.4	5.987
Test 6	9.2	14.4	94.4	6.545
Test 7	8.3	14.4	104.3	7.228
Test 8	7.4	14.4	116.1	8.049
Test 9	6.5	14.4	131.6	9.117
Test 10	5.5	14.4	151.6	10.508
Test 11	4.6	14.4	178.1	12.344
Test 12	3.7	14.4	218.4	15.113
Test 13	3.2	14.4	246.6	17.088

Following the expressions derived as given in Equation 20 through Equation 23, it is obvious that the relationship between the R and Z values should be inversely proportional. Therefore, the results of the calibration tests are curve-fitted to verify this finding and to explore the expression of the relationship function. The result is shown in Figure 10; and the graph shape confirms the inverse proportionality of the relationship function which is given in Equation 25.

$$R = \frac{63.3}{(0.4732 + Z)} \left(\frac{pixel}{mm} \right) \quad (25)$$

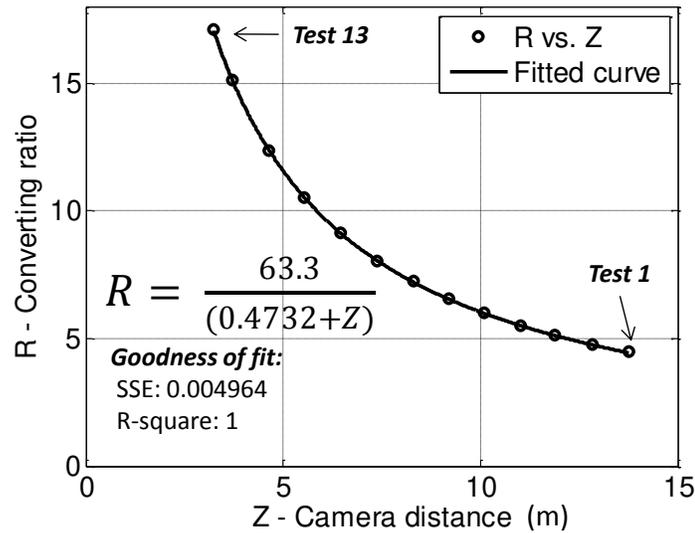


Figure 10. The relationship between the conversion ratio R vs. the camera distance of Z obtained for a Canon VIXIA HF R42 camcorder

Laboratory Verification

The proposed computer vision-based monitoring demonstrated first on a small-scale bridge at the University of Central Florida (UCF) Structures Lab. The bridge named UCF 4-Span Bridge Model consists of two 304.8 cm main continuous spans and two 120 cm approach spans. The model deck is 120 cm wide 3.18mm steel sheet compositely connected to two HSS 25x25x3 mm steel girders separated 60.96 cm from each other as shown in Figure 11. As a part of various SHM studies at the UCF, there are a large number of sensors attached on the model such as strain gages, accelerometers, LVDTs, tilt-meters, and Fiber Bragg Grating (FBG) sensors. Moreover, some common bridge damage scenarios can be replicated by changing boundary conditions at the supports or altering the local stiffness of the girders by losing some bolts to reduce composite action of the model elements. To simulate traffic loading, small-scale vehicles are deployed back and forth on the bridge deck. In this Chapter, displacements of the small-scale bridge due to traffic

induced loading from a small-scale vehicle are obtained by utilizing the non-target vision-based displacement measurement method.



Figure 11. The 4-span bridge at the UCF Structural Laboratory

Experiment design

For verification purposes, the monitored region is selected as close as possible to the location of pre-attached sensors (LVDT and accelerometer) located under the main girder at the two-fifth-span point (Figure 12). There is a data acquisition system for simultaneously collecting data from all sensors and the camera. To confirm reliability and consistency of the proposed method, a total of six (6) tests is conducted by altering small-scale vehicle weights and speeds, as well as locations and angular orientations of the camera as detailed in Table 2. A small-scale vehicle is driven over the bridge deck following pre-defined loading configurations and speeds. As the small-scale vehicle is actually a dynamic load, it induces not only static displacements but also dynamic vibrations on the structure. Such obtained displacements and vibrations are presented and verified in the following sections.

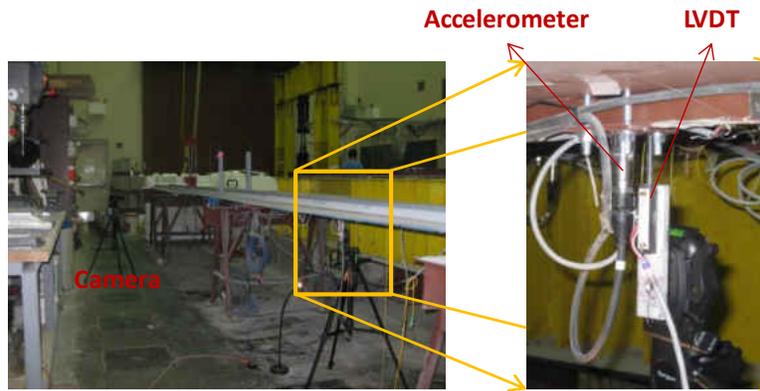


Figure 12. Measured location and experimental setup

Table 2. The test specifications

	Speed ^a (m/s)	Weight ^a (kg)	Distance ^b (m)	Angle ^b degree
Test 1	0.32	12.3	2.04	0
Test 2	0.26	12.3	3.74	0
Test 3	0.27	12.3	3.75	6
Test 4	0.30	12.3	2.07	11
Test 5	0.79	12.3	2.04	0
Test 6	0.37	5.30	2.04	0

^a Speed and Weight of the small-scale vehicle

^b Distance value Z from camera to the measured location; and angular orientation of camera.

Laboratory results and discussion

The displacement results at the measurement location of all tests are shown in Figure 13. The sample rates of all data sets measured by both LVDT and the proposal vision based method are 30 Hz. In each of the graphs, both raw displacement data collected from the LVDT and the proposed vision-based method are synchronized and illustrated in a comparative fashion.

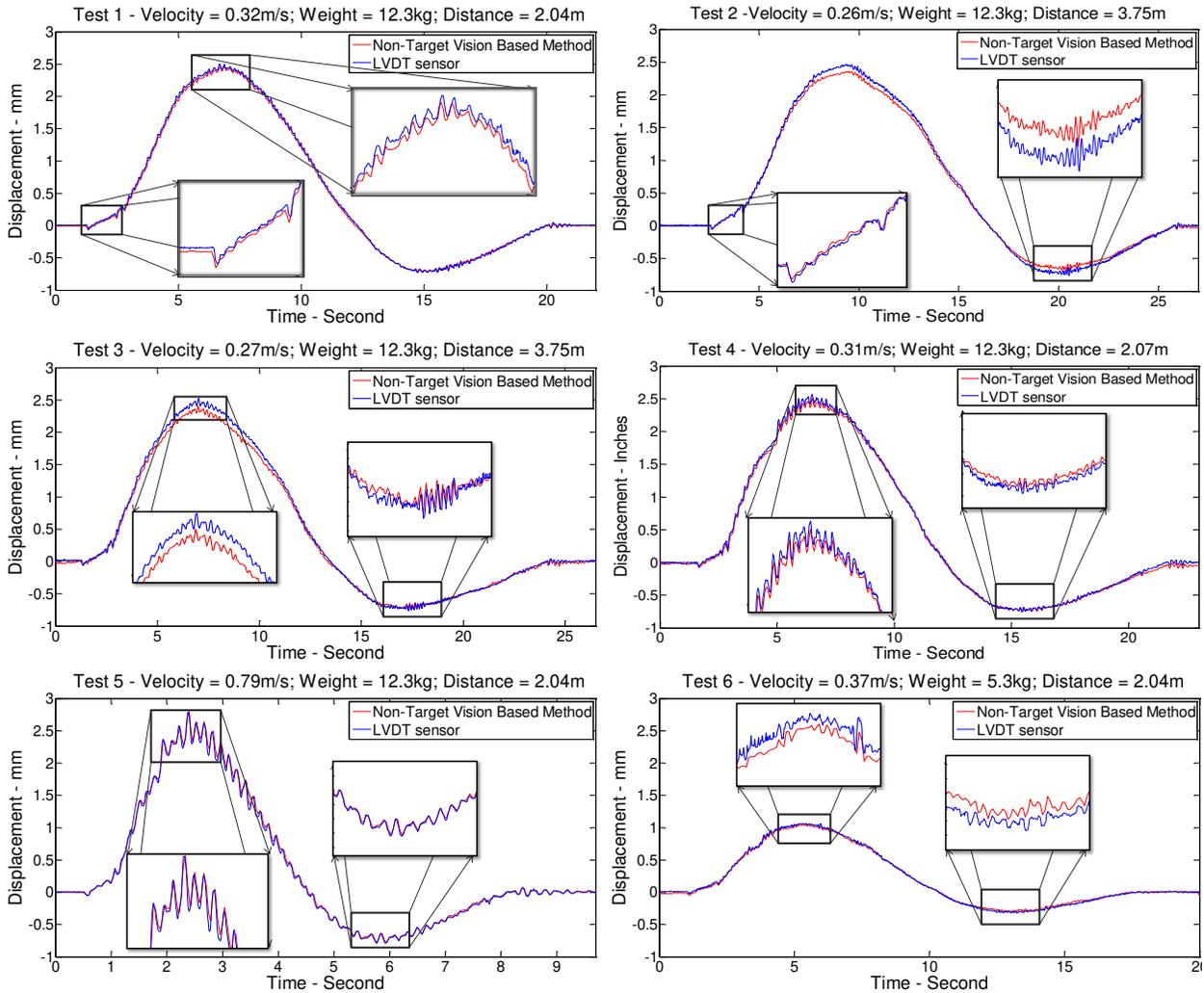


Figure 13. Comparison of displacement results by using the LVDT and the proposed method

It is seen that although the experimental setups are different among the tests, displacement values determined from the non-target vision-based method highly correlate with the results obtained from the LVDT sensor. This observation is even confirmed strongly at every peak of the vibration data as shown in the insets in Figure 13. To measure the correlation behavior between two datasets, the correlation coefficient factors (ρ) are determined by Equation 26 for every experiment. In addition, a statistical measure named determination coefficient (*R-squared*) factors

is computed to determine how well the two datasets match together. The *R-squared* value can be calculated using Equation 27.

$$\rho = \frac{|\sum_i(d_L(i) - \mu_{d_L}) \times (d_v(i) - \mu_{d_v})|}{\sqrt{\sum_i(d_L(i) - \mu_{d_L})^2} \sqrt{\sum_i(d_v(i) - \mu_{d_v})^2}} \quad (26)$$

Where d_L and d_v are the dynamic displacement values extracted by filtering out the static component of the raw data from the LVDT sensor and the proposed vision-based method, respectively; and μ_{d_L} and μ_{d_v} are the mean values of two above datasets. The values of ρ vary from 0.0 to 1.0; and $\rho = 1.0$ shows perfect correlation whereas $\rho = 0.0$ indicates no correlation between two datasets.

$$R^2 = 1 - \frac{\sum_i(r_v(i) - r_L(i))^2}{\sum_i(r_v(i) - \mu_{r_v})^2} \quad (27)$$

Where r_L and r_v are the raw displacement values obtained by the LVDT and proposed vision-based method, respectively; and μ_{r_v} is the mean value of the raw dataset determined by the proposed method. The values of R is from 0.0 to 1.0; and the R value of 1.0 implies the perfect similarity between two datasets.

The comparison results between two displacement datasets obtained from a classical sensor and a new vision-based method are described in Table 3. As seen in Table 3, maximum and minimum static displacements of the bridge model are consistent from *Test 1* to *5* based on the responses from the same small-scale vehicle weight (12.3 kg). However, the lighter weight vehicle (5.3 kg) being utilized at *Test 6* induces smaller displacement amplitudes and the ratio of displacement amplitudes is similar to the weight ratio. It is also seen that the correlation coefficient ρ values (from 0.929 to 0.971) and the determination coefficient *R-squared* values (from 0.9987

to 0.9998) are close to 1.000, which indicate a very high correlation and similarity between the two compared datasets. Hence, the displacements obtained by the non-target displacement measurement method show comparable and accurate results when under laboratory condition.

Table 3. Correlation and error analysis for the proposed method

	Vision-based Method		LVDT		Error		Corr. (ρ)	R-squared
	Max Disp. ^a	Min Disp. ^a	Max Disp. ^a	Min Disp. ^a	Max Disp. ^a	Min Disp. ^a		
	(mm)	(mm)	(mm)	(mm)	(%)	(%)		
Test 1	2.45	-0.71	2.47	-0.72	0.8%	1.4%	0.969	0.9997
Test 2	2.36	-0.64	2.41	-0.67	2.1%	4.5%	0.971	0.9998
Test 3	2.37	-0.68	2.45	-0.69	3.3%	1.4%	0.961	0.9995
Test 4	2.46	-0.70	2.48	-0.71	0.8%	1.4%	0.929	0.9995
Test 5	2.51	-0.74	2.50	-0.74	0.4%	0.0%	0.951	0.9987
Test 6	1.06	-0.30	1.07	-0.31	0.9%	3.2%	0.938	0.9988

^a The static maximum and minimum displacement values after filtering dynamic behavior.

Furthermore, the dynamic characteristics of the 4-span bridge model can be captured by analyzing the dynamic component of the raw displacement datasets. By FFT-transforming the dynamic displacements from time domain to frequency domain, the natural frequencies of the bridge model can be identified as shown in Figure 14 at 4.9 Hz, and 6.9 Hz. In the meantime, vibration response of the bridge model is collected by using an accelerometer attached at the same location captured by the camera. Figure 14 also shows the raw data recorded by using that accelerometer; from which the first and second natural frequencies of the structure can be detected. The identified frequencies from accelerometer data are 5.2 Hz, and 7.0 Hz that closely match to the frequencies of 4.9 Hz, and 6.9 Hz identified using the proposed vision-based method. However,

in spite of the good match for the first two frequencies, higher frequencies of the bridge structures can only be identified using accelerometer data.

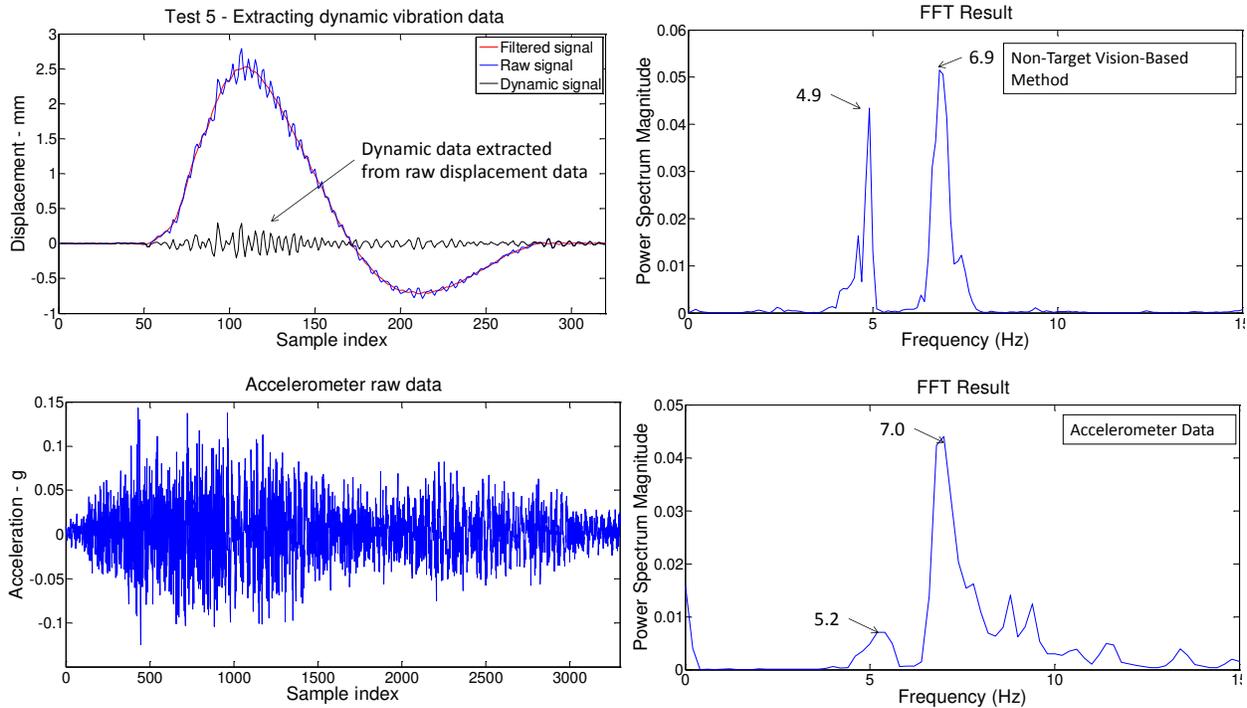


Figure 14. Comparison of identified natural frequencies using the proposed method and the accelerometer data

Through the results of the non-target vision-based displacement measurement method described in this section, it is seen that the proposed algorithm can not only obtain static displacements but also identify dynamic characteristics of the 4-span bridge model. The outcomes from different experimental setups confirm the accuracy of the proposed method consistently in the laboratory. For further verification, real-life structures are conducted and the corresponding results are presented and discussed in the following sections.

Field verification on an elevated guideway for APM systems

An automated people mover (APM) system mainly includes four basic components: trains, guideways, stations and a control system. For the system presented here, trains are designed to run on a viaduct system comprised of multiple span bridges that connect four airside terminals as shown in Figure 15. Operating every 2 minutes from 5am to mid-night, maintenance of the skybus system is critical to guarantee the safety of passengers. As a part of the APM guideway rehabilitation project, some spans of the viaduct are monitored to verify the behavior of the structure after retrofitting. Because of the access and space limitation (Figure 15b), all monitoring equipment are designed wirelessly including accelerometers, strain gages, and cameras in order not to interfere with the busy under viaduct traffic. Although there are different types of sensor data available from this monitoring project, only displacement and vibration responses of the viaduct are obtained using the proposed vision-based method under train loadings and the results are verified using sensor-based measurements.



Figure 15. (a) Trains running on the viaduct, (b) Runway under the viaduct

Considerations for changing ambient illumination

In general, the most common obstacles that affect both the quality of images and the accuracy of obtained results when conducting a vision-based monitoring are 1) far distance from the camera to the measurement locations, and 2) changing ambient illumination when collecting video clips. In this monitoring study, especially changing illumination issues had to be resolved. For the setup of the experiment, the camera is focused on a measurement location near the midpoint of a main girder from a distance of 11.5 meters away from the camera location, where a wireless accelerometer (AS1-ACC25) is attached (Figure 16).

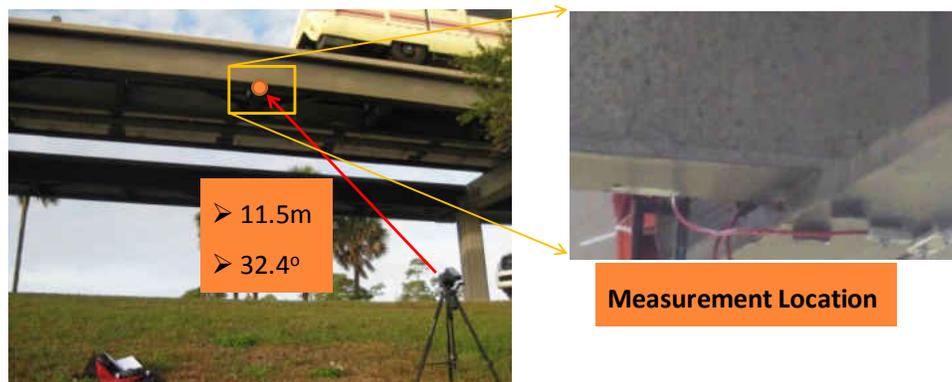


Figure 16. Measurement location and camera setup

Image sequence of the measurement location is captured using the camera when trains pass over the monitored span. Highly changing of ambient illumination and low contrast of images due to shade of the trains on the measurement location are the challenges to be solved (Figure 17). It is difficult to address these problems by using common vision-based displacement measurement methods, which are based on the image correlation algorithms. Besides, the correlation approach is very sensitive to both illumination changes and low contrast of photography. By utilizing the proposed method, the matching key-points algorithm automatically selects the strongest key-points from an appropriate illumination region to match. In this monitoring study, the measured

positions are captured under highly changing of light illumination as well as dark condition (due to under the bridge deck), the false-matching rate is a high value. Then, the outlier discarding algorithm is performed to successfully obtain the best matches as illustrated in Figure 18.



Figure 17. Highly changing illumination among image frames due to shade of the trains

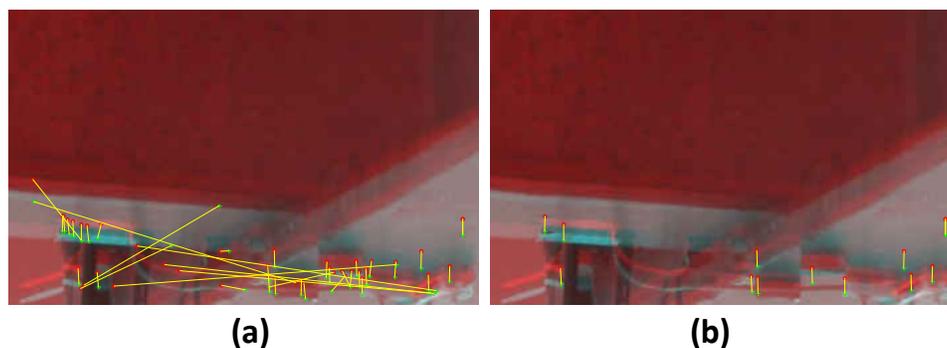


Figure 18. (a) Automatically matching the strongest key-points for low change of illumination region, (b) The best matches after utilizing the outlier discarding algorithm

Obtaining Structural Displacements and Identifying Dynamic Frequencies

Figure 19 shows the displacements of the measurement location beginning from the time of the trains approaching the monitored span to the time of their exit. A total of three data tests were collected that illustrate the consistency of the general behavior such as the deflection pattern, maximum displacements (~ 12 mm), and the response time (~ 4.27 seconds). Although there is not

any LVDT information to verify the maximum displacements in a comparative sense, the deflection pattern and the time of structural response are observed to be similar to strain pattern and response time (~4.32 seconds) acquired from a strain gage mounted at the same location.

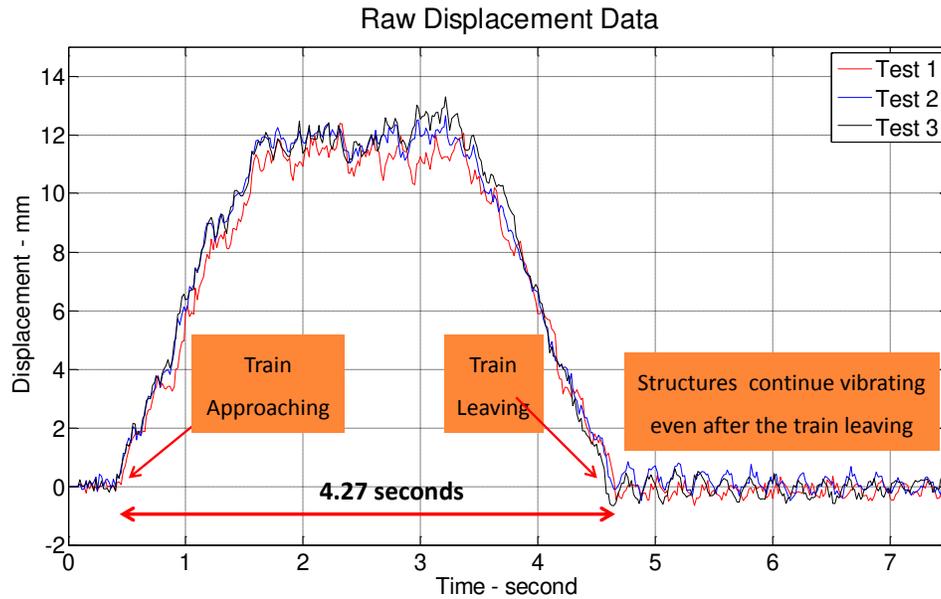


Figure 19. Raw displacement data of a measurement location on obtained using the proposed method

Modal frequencies of the main girder can be identified using the dynamic component of the displacement data extracted from the raw displacement history by filtering out the static response. In addition, acceleration responses of the same girder are collected by total of ten (10) accelerometers attached at different locations on the girder (including the one at the same camera capturing location, AS1-ACC25). Dynamic responses from both displacement and acceleration data are shown in Figure 20. Natural frequencies found via the processing of both camera and accelerometer data are presented in Figure 21. While one can observe some minor differences in the identified frequencies, a considerable number of the dominant frequencies identified using two different approaches match. The errors between two sets of frequencies are calculated in Table 4

indicating errors less than 2.5%. The comparison result confirms that the proposed vision-based method works efficiently even in case of a real-life structure under difficult lighting condition.

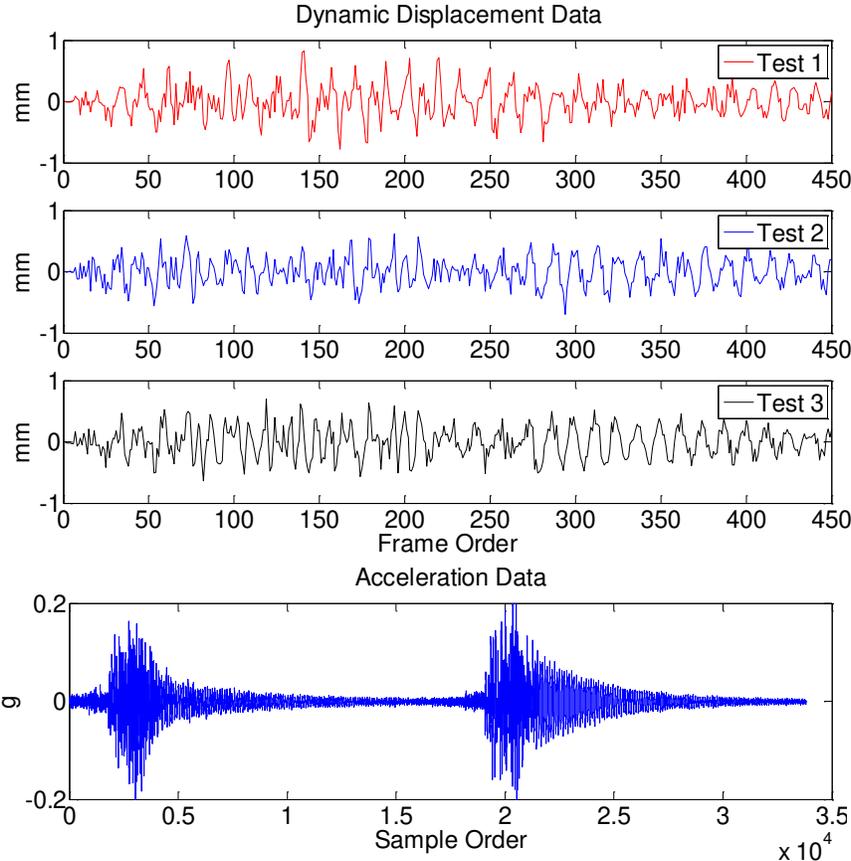


Figure 20. Dynamic responses of the main girder extracted by the proposed method and an accelerometer (AS1-ACC25)

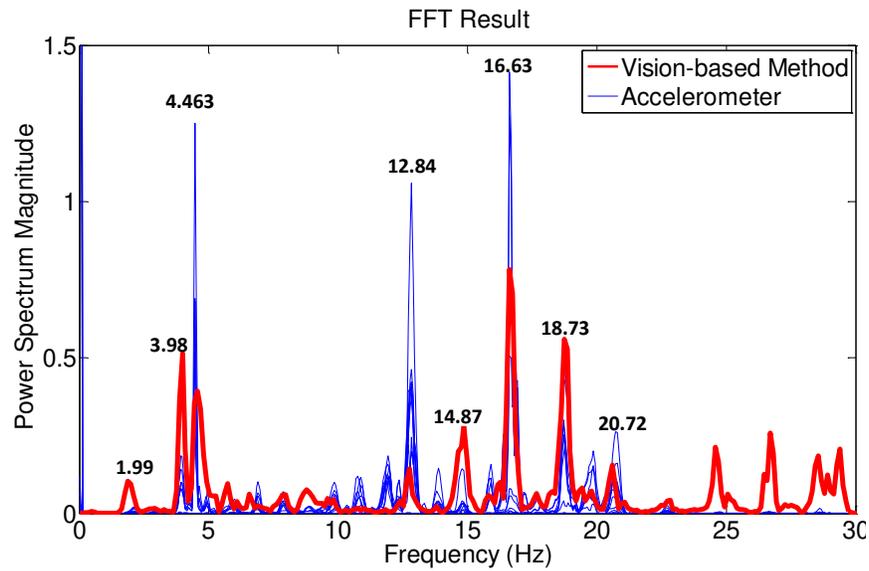


Figure 21. Comparison of natural frequencies of the girder identified by the proposed vision-based method and accelerometers

Table 4. Comparison of natural frequencies

	Vision-based (Hz)	Accelerometer (Hz)	Error (%)
1 st Freq.	3.98	3.97	0.18
2 nd Freq.	4.57	4.46	2.44
3 rd Freq.	12.76	12.84	0.62
4 th Freq.	14.87	14.79	0.54
5 th Freq.	16.62	16.63	0.06
6 th Freq.	18.73	18.77	0.21
7 th Freq.	20.61	20.72	0.53

Field implementation on a Stadium Structure

Assessment of civil engineering structures that are densely occupied by people such as stadiums, theaters, airports, stations etc. are significantly important not only for safety but also for

serviceability and human comfort reasons. For such structures, sport stadiums deserve special attention due to having to serve under variable harsh loading conditions induced by unpredictable and excited fans. Some structural response issues for monitored stadiums during games are excessive deflections, high vibration levels, and resonance problems, which have significant effects on human comfort experience (Caprioli, Reynolds, & Vanali, 2007; F Necati Catbas, Gul, & Sazak, 2009; Salyards & Hanagan, 2009). It becomes very important to determine the response characteristics (deflections, vibration levels, human comfort indices) during the games with practical and accurate methods without interfering the crowd and the infrastructure owners. In this section, the non-target vision-based displacement and vibration monitoring is demonstrated based on the methodologies provided previously for verification as a part of a football stadium monitoring project. Completed in 2007, this stadium is a steel structure with approximately 45,000 seating capacity (Figure 22). Although it was designed and constructed recently, it exhibited considerable vibration levels especially at the sections of the highly active team supporters and fans, and consequently, these sections were retrofitted later on. Before and after the retrofit, an SHM study was performed particularly at these sections of the structure where the most excessive responses are most likely to occur due to the more excited crowd.

The non-target vision-based method is implemented under altering illumination conditions, different camera locations (distances and angles), and camera frame rates. The dynamic displacement response data to different crowd reactions are also analyzed in frequency domain to identify structural frequencies of the stadium. Those responses and identified dynamic characteristics are compared to ones acquired from conventional sensors including accelerometers and LVDTs.



Figure 22. The Football Stadium Monitored During Games

Stadium Displacements: Raw Data and Verification

The responses of the aforementioned football stadium are investigated using the data collected during the games. The most significant movements of the crowd are observed in reaction to some typical events during the American football game such as "kickoff", "touchdown", "interception", "fumble" etc. and in return, the excitement of interest is generated by in forms of cheering, bouncing, stamping, and even jumping. To verify the non-target vision-based monitoring method, the tests are implemented on a supported girder of the grand stand (a retrofitted W16x40 steel beam) right underneath the active crowd seating. There are total number of two (2) LVDTs and four (4) accelerometers attached on the bottom side of the girder as illustrated in Figure 23 for the verification of the proposed method for structural displacement measurement and frequency identification.

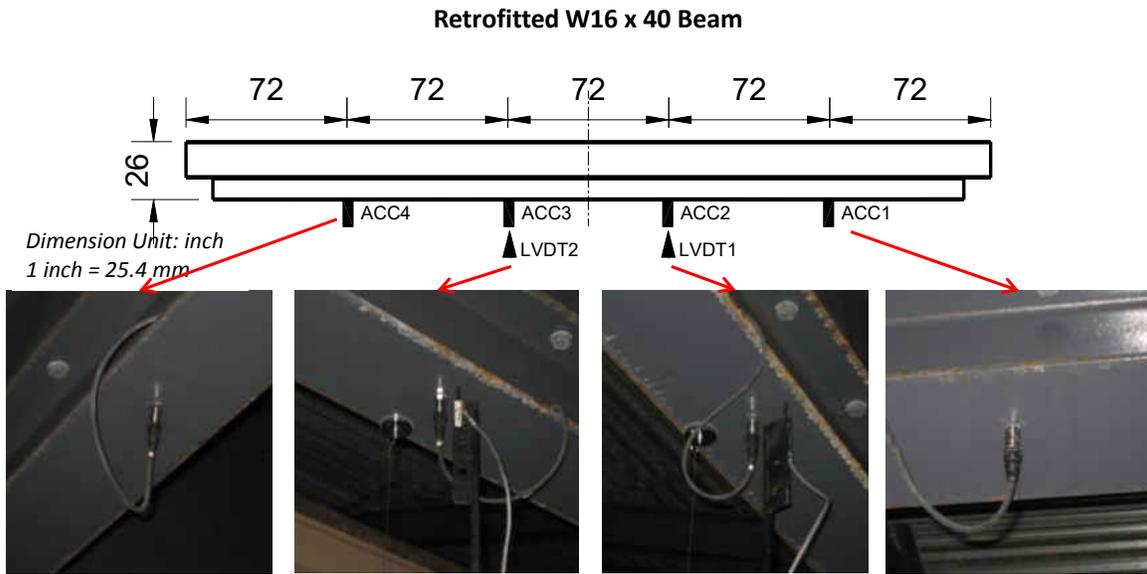


Figure 23. Experimental setup on the supported beam



Figure 24. Camera placement on a tripod

Although several experiments were conducted in the stadium, only four (4) of them are presented in this section; and each study stands for a certain test scenario. The details of the test scenarios are shown in Table 5.

Table 5. The test scenarios

	Distance ^a (m)	Angle ^a (degree)	Frame Rate (Fps)	Light Condition
Test 1	7.16	19	30	Night, Dim
Test 2	10.21	13	30	Night, Dim
Test 3	11.58	11	30	Day, Dim
Test 4	5.85	23	60	Day, Dim

^aDistance and angle of the camera

The captured video clips for all the tests have been processed following the flowchart in Figure 2. Meantime, the sensor data was acquired by a National Instrument DAQ system. Figure 25a illustrates raw displacement history obtained using proposed computer vision method as well as using *LVDT 1* at the same location (see Figure 23). These data sets were collected when the home team scored the 2nd touchdown, and subsequently, the crowd started to stamp and jump with the popular song called "*Zombie Nation*", which lasted for about 30 seconds. To visually compare the results of the two methods, a 15 second time window is picked and illustrated with both results overlapped (Figure 25b). This plot confirms that the outcomes of the two methods are similar in terms of displacement levels, vibrating pattern, and the matching of the peak displacements.

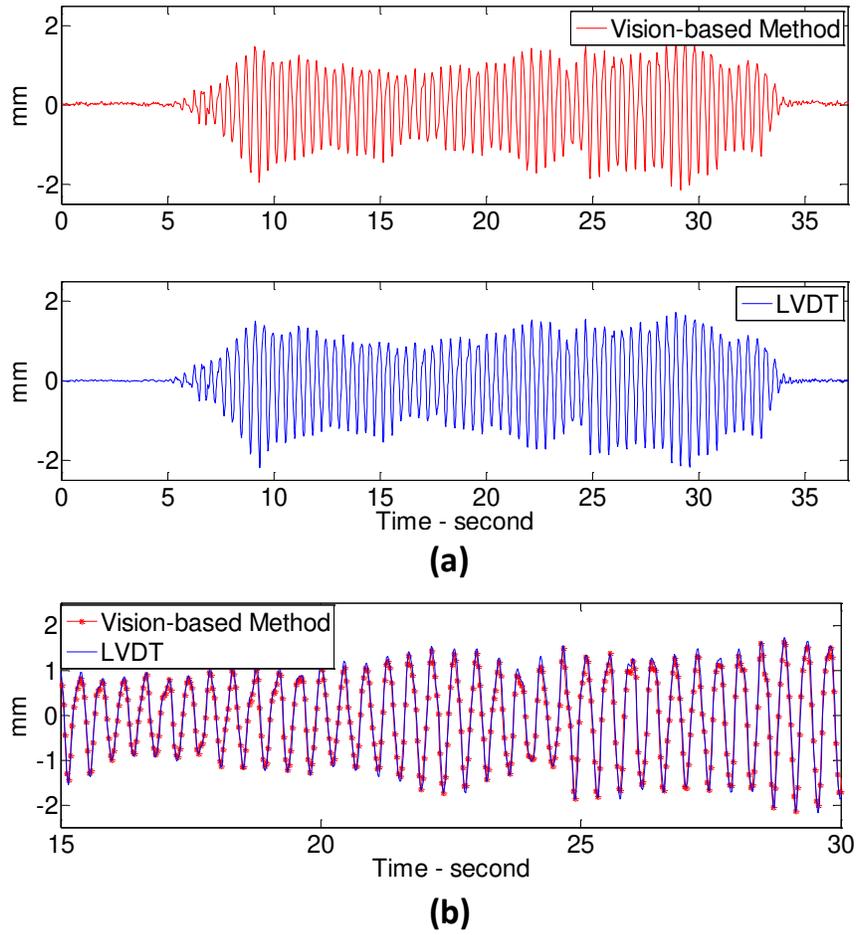


Figure 25. (a) Test 2 displacement datasets, (b) An zoom-in window from Test 2

Another dataset is recorded for a much longer time period (about 260 seconds) at the beginning of another game. In this event, the recording starts with spectators cheering and jumping for 30 seconds when the home team takes position on the field until its players run for the kickoff that excites the crowd in the form of jumping in accompaniment with a special song for the home team. All these observations are clearly captured by all sensors as well as the camera of the vision-based monitoring system (Figure 26a). In this case, two time-windows (20th sec - 35th sec and 220th sec - 235th sec) are employed for detailed analysis as presented in Figure 26b. The two data sets showing both vision-based method and *LVDT* measurements almost perfectly match.

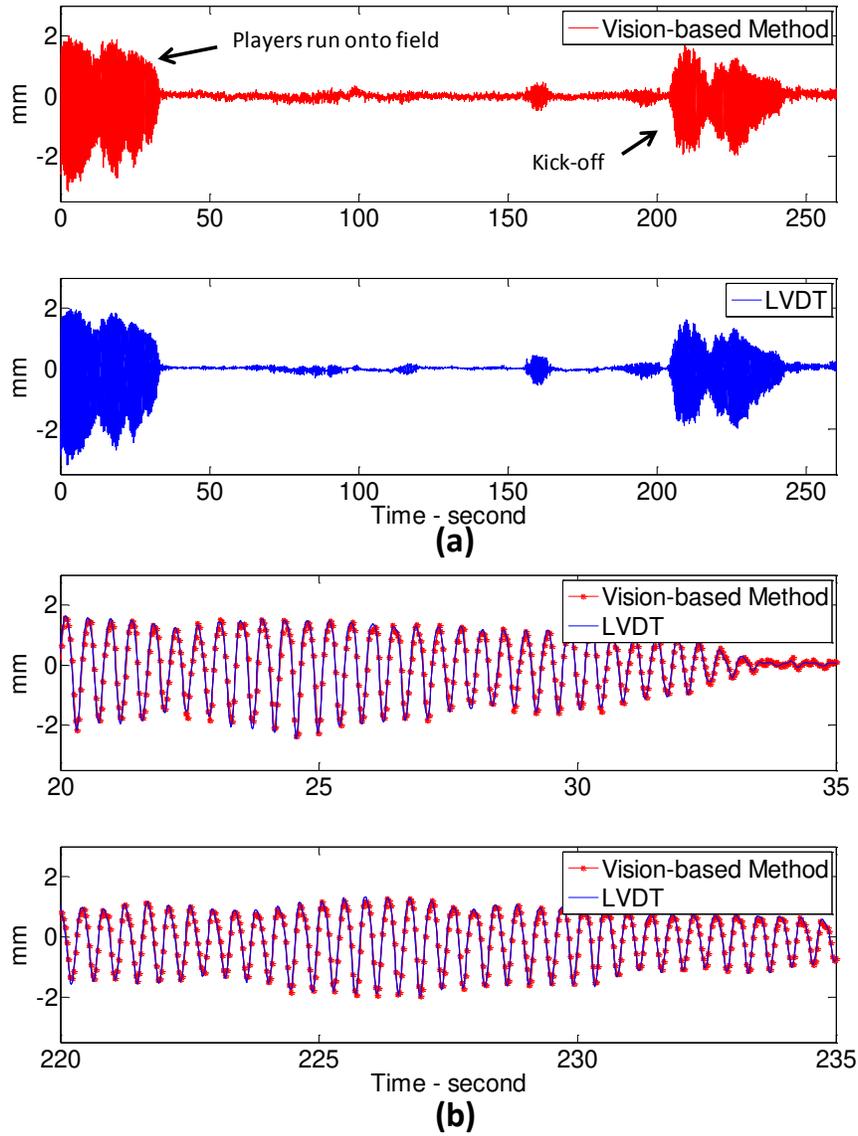


Figure 26. (a) Test 3 displacement datasets, (b) Two zoom-in windows from Test 3

In addition, quantitative analyses are performed to verify the correlation between *LVDT* and vision-based results. The correlation coefficient factor (ρ) is determined by the following formulation (Equation 26). Moreover, error analysis of these two displacement datasets are also performed and quantified using determination coefficient (*R-squared*) factors. The value of *R*-

squared factor can be calculated by utilizing Equation 27 and it indicates how well these two datasets fit together.

The correlation and error analysis results are described in Table 6. As seen in this table, although the Tests 1 and 4 were executed at closer distances, the results from these tests are slightly lower than the ones from the Tests 2 and 3, in which the angles of the camera orientation are smaller. It can be concluded that the results obtained by the proposed vision-based method are more sensitive to angle of the camera than the camera distance for the ranges given in these tests. In summary, the correlation coefficient ρ (from 0.9689 to 0.9887) and the determination coefficient R-squared (from 0.9368 to 0.9775) are very close to 1. Those values imply that the stadium displacements determined by the non-target displacement measurement method show reliable results when compared with the data obtained from the LVDTs for all four tests.

Table 6. Correlation and error analysis

	Distance (m)	Angle (degree)	Corr. (ρ)	R-squared R^2
Test 1	7.16	19	0.9880	0.9426
Test 2	10.21	13	0.9877	0.9751
Test 3	11.58	11	0.9887	0.9775
Test 4	5.85	23	0.9689	0.9368

Identification of the Modal Frequencies of the Structure

Dynamic characteristics of the stadium are determined by analyzing displacement data from the vision-based method induced by crowd excitation. The natural frequencies of the supported beam are identified by analyzing the proposed method data and compared with the

accelerometer data in frequency domain as shown in Figure 27. Although accelerometer data shows additional frequencies for the higher frequency band especially beyond 10 Hz, it is seen that the first three natural frequencies identified by two different methods are perfectly matching at 2.37 Hz, 4.75 Hz, and 6.62 Hz, respectively. This observation confirms the quality of vibration data acquired from the proposed method, and enables a potential complement for the most common sensors deployed in SHM including LVDTs and accelerometers for the type of real-life applications where access might be an issue. In addition, data can be collected from any locations with an engineer or inspector.

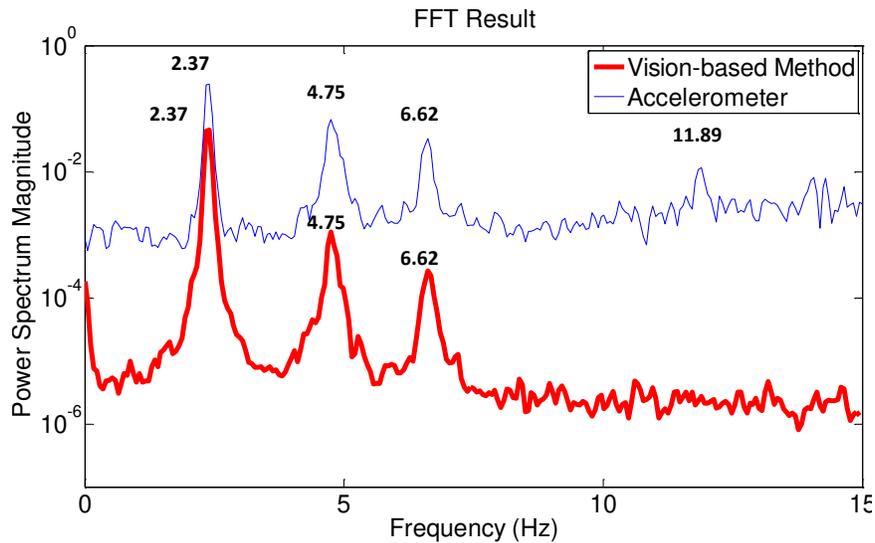


Figure 27. Comparison of natural frequencies of the supported beam identified by the proposed vision based method and processed acceleration data

Accuracy of the Proposed Method

Unlike classical measurement approaches, in which sensor accuracy is provided by its manufacturer with a roughly stable value, accuracy of a vision-based measurement system depends on several parameters. In such parameters, the distance from cameras to measurement positions

possibly is the most critical one. In this section, an approach is presented to evaluate vision system accuracy based on statistical analysis of error in data. The proposed vision-based monitoring is carried out with a Canon VIXIA HF R42 camcorder to measure displacements of a perfectly stationary object (a 20 cm diameter of steel round barrier in a laboratory) for a series of pre-defined distances of Z_{accu} (Figure 28). While the real displacements of the stationary barrier must be zero under ambient condition, the non-zero displacements obtained by the proposed vision-based monitoring must be errors. Just like the camera calibration procedure, evaluation of the vision system accuracy is one time, independent deployment.

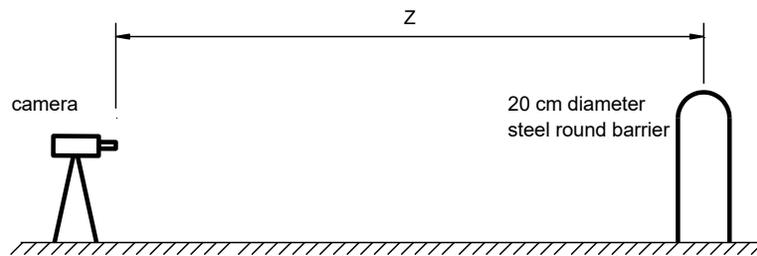


Figure 28: Experimental setup for the method accuracy evaluation

Figure 29a shows the distribution of errors for a particular Z_{accu} distance of 2.9 m. The distribution can be assumed as normal with the mean value of zero and the standard deviation of 0.0049 mm. Results from the other accuracy evaluation tests conducted at distances for 2.9 m, 8.9 m, and 13.5 m are also illustrated in Figure 29b. Although the mean values of those distributions are zero, the standard deviation values decrease for closer measurement distances.

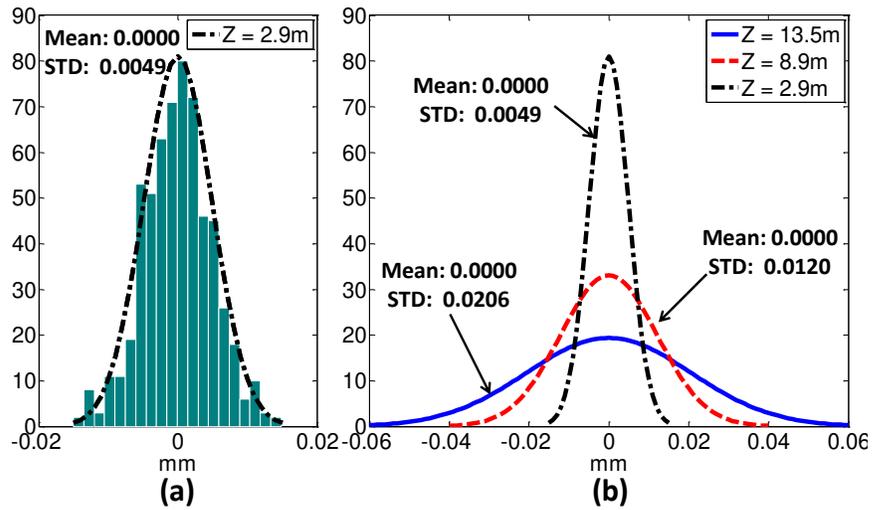


Figure 29. Distributions of error data obtained from several accuracy evaluation tests

By defining an accuracy interval for a measurement equal to ± 2 times of a standard deviation of the measurement error distribution (at a level of confidence of 95%), the accuracy of the proposed vision-based monitoring can be determined corresponding to the pre-defined distances of Z_{accu} ; and the results are demonstrated in Figure 30. It is seen that at a distance of 3 m, the accuracy of the proposed measurement is ± 0.01 mm; however, the accuracy becomes ± 0.04 mm for a measurement distance of 13 m. Moreover, it is also observed that the relationship between the accuracy of the proposed vision-based monitoring and the camera distance is a linear function, which can be represented by the fitted line as shown in Figure 30.

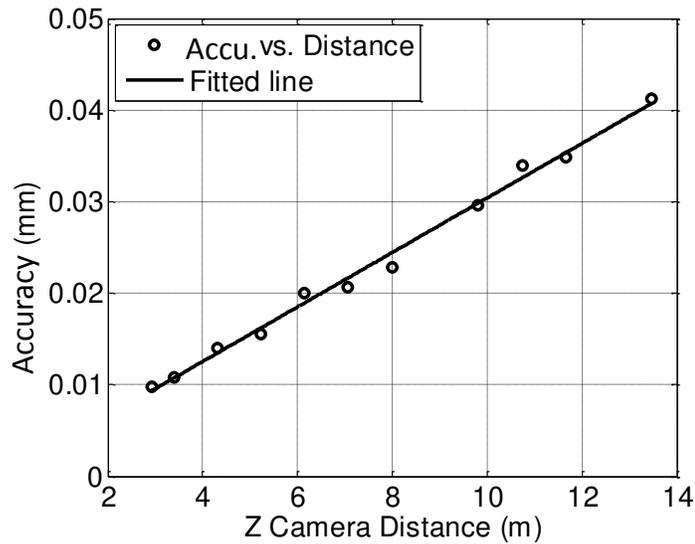


Figure 30. The relationship between the accuracy of the proposed measurement vs. the camera distance of Z obtained for Canon VIXIA HF R42 camcorder

Conclusions

This study demonstrates a completely contactless SHM system for obtaining displacements and vibrations of structures using a low-cost camera and computer vision techniques. The new method makes it possible to obtain not only static displacements but also displacements with vibration characteristics, from which structural frequencies are identified. The advantage of this proposed method is that it does not require any type of physical targets attached on structures which are commonly required by other vision-based methods. Non-contact monitoring is achieved by means of a new type of virtual markers instead of physical targets. The key-points of measurement locations are extracted by using robust computer vision algorithms, and their characteristics show a potential ability to replace physical targets. Key-point matches among image frames are further improved by using an outlier detection algorithm to discard false matches.

Furthermore, a calibration procedure is developed to calculate the unit conversion ratio R between the image and the world coordinates when conventional targets no longer exist.

The proposed method and framework are implemented and verified in a laboratory apparatus and two real-life structures under different experimental conditions including altering light conditions, different camera locations (distances and angles), and camera frame rates. Since this method does not require any type of attachment, its implementation is observed to be much more convenient (e.g. faster and easier) than other conventional sensors employed on the same structure. For verification purposes, the displacement data sets for different influences are obtained using the proposed method to be compared to conventional LVDT data. The comparison results are validated by means of visual inspection and some statistical measures such as correlation coefficient ρ and the determination of *R-squared* coefficient. Moreover, dynamic frequencies are also identified from the dynamic displacement time histories as the camera can capture images at speed rates of 30 Hz and 60 Hz. Identified natural frequencies are compared to the frequencies obtained using acceleration data. The comparison of these two aforementioned frequency values shows perfect matching in most of the cases. The real-life study indicates that important structural responses can be determined using non-contact vision-based monitoring method presented in this paper. Although the proposed method provides improvements and advantages for practice including completely non-contact implementation, low-cost hardware and quite accurate results for defined measurement ranges and conditions, some related issues such as data storage requirement for clips and images, processing time for image data, and limitation for horizontal displacement measurement need to be considered.

CHAPTER THREE: DETECTION AND LOCALIZATION OF VEHICLES ON A BRIDGE USING COMPUTER VISION APPROACH

Introduction

Loading estimation is a very important task as for both designing a new civil engineering project and the assessment of existing structures. In the field of Structural Identification (St-Id), these loads are commonly utilized as input parameters in conjunction with output structural responses for identifying damage. Additionally, the loads and corresponding responses are also continuously updated to obtain calibrated FE (Finite Element) models for numerous aims such as safety evaluation, damage detection, and eventually prediction of the remaining life of structure. In general, bridge loading effects might be exemplified as wind, temperature, earthquake, pre-stressed tension, vehicles etc. However, the influence of vehicle loading is strongly being interested in Bridge Health Monitoring (BHM) since it can be easily controlled by inspectors. In a common monitoring study, the pre-weighted trucks are located or crawled on the monitored bridge following predefined configurations. The axle weights of the trucks are obtained by a weight-scale or a Weight-In-Motion (WIM) system (in some exceptional cases), while their locations are determined using labor (Figure 31). This manual process requires bridge closures as well as being time consuming and labor-intensive that makes bridge inspection and monitoring costly and ineffective.



Figure 31. Pre-weighted trucks deployed in a common bridge monitoring study

The goal of this section is to propose an alternative approach employing traffic vehicles instead of using the testing trucks for BHM, thereby mitigating some previously noted shortcomings of the current practices. The proposal process is immensely based on vision technologies for obtaining information about the traffic passing on the bridge deck, such as vehicle weights and positions. The protocol hardware simply consists of a surveillance camera system with the aim of capturing traffic scenes on the bridge. Subsequently, the acquired images are analyzed by a computer vision software package for firstly detecting and then classifying traffic vehicles into classes. The weight-distribution of a particular vehicle class can be developed from manufactures specifications coupling with weight-in-motion (WIM) database. Thus, the weight of a classified vehicle is assigned equal to its class weight-distribution that has been pre-developed. In addition to estimating vehicle weights, the detected vehicles are located in terms of geometric transformation between the image coordinate system (acquired by the surveillance camera) and the world coordinate system by which the bridge deck is defined in real life. Once the vehicle information is obtained, this data (instead of manually obtained data from the testing trucks) is transmitted to a St-Id system for structural assessment. Data acquisition without the constraint of bridge closure saves time and labor work in bridge monitoring practices, and also mitigates

uncountable hours of traffic jam and detour. The proposal in this chapter is developed by means of numerous advanced computer vision algorithms involving object detection, classification, and localization, which have been substantially progressed in recent years.

In computer vision, object detection and tracking are the most appealing challenges since those algorithms are fundamental techniques for other diversity implementations (Javed & Shah, 2002; Yilmaz, Javed, & Shah, 2006). Related to vision based analysis of traffic vehicles, it is a matter of raising importance in Intelligent Transportation Systems (ITS). Following the review of computer vision techniques for traffic analyses (Buch, Velastin, & Orwell, 2011), it is seen that hundreds of research papers have been published in the last decade in the context of ITS field. These publications mainly cover detection, tracking, and classification of vehicles for multiple aims such as counting vehicles, detection of traffic violations (illegal turns, over speed limit, etc.), automatic number plate recognition (ANPR), traffic control and incident detection. In this pool of studies, vehicle detection has been put in much more effort due to the grants allocated for subsequent tasks of detected vehicle information in a number of traffic analysis frameworks, such as vehicle tracking, classification, and localization. Generally, vehicle detection algorithms are categorized into two groups named as motion-based and appearance-based approaches.

The first group of vehicle detection, motion-based methods, identify vehicles by analyzing frame sequences of video clips (Cedras & Shah, 1995). Since traffic flows, positions of vehicles on a particular image shift to other locations on the next image. A fundamental computer vision algorithm based on foreground estimation and segmentation is implemented for numerous vehicle detection studies. In that method, foreground regions are defined as moving objects, which do not belong to the scene (so-called the background). Given an image of highway traffic, detected foreground regions are certainly supposed as hypothesized vehicles. These foreground regions are

commonly obtained by means of the frame differencing or the background subtraction algorithm. An early research focusing on the recognition of traffic vehicles is presented by Gupte et al. (Gupte, Masoud, Martin, & Papanikolopoulos, 2002). As described in the paper, a self-adaptive background subtraction algorithm, that could extract foreground regions under changing illumination condition of background, is employed. Using the similar process to the one presented in Gupte's study, Huang and Liao try to address the occlusion problem by predicting merged borders of foreground regions based on the velocity estimation of vehicles (Huang & Liao, 2004). Due to the authors being able to identify occlusion occurrences, foreground regions could be divided into multiple vehicles making the detection more robust. A complete framework for traffic analysis is introduced to comprise studies for detection, tracking, classification and also counting vehicles on highways (Rad & Jamzad, 2005). In this implementation, the authors employ the closing and opening morphological methods to get more robust foreground regions. Hsieh et al. propose an approach for solving a common challenge called shadow problem when utilizing background subtraction algorithm (Hsieh, Yu, Chen, & Hu, 2006). Since vehicles are regularly located on inside lanes, the authors could determine shadow and occlusion happenings by lane-dividing lines. A new feature named 'linearity' is developed in addition to dimensional features yielding better results from the vehicle classification. Besides, to address the occlusion, shadow and other drawbacks of the background subtraction method, Su et al. propose a novel and effective framework named collaborative background extraction algorithm for detection and tracking multiple vehicles (Su, Khoshgoftaar, Zhu, & Folleco, 2007). Following this method, a detected vehicle could be firstly consolidated and then be separated from neighboring vehicles based on collaborations among them along a sequence of images. Furthermore, shadows of vehicles and falsely detected foregrounds are successfully rejected as the outliers of background. Besides the

previously mentioned shortcomings, the background subtraction method suffers from intensive computation as well. Therefore, some other studies focus on the optimization of computational efficiency (Z. Kim, 2008; Vargas, Milla, Toral, & Barrero, 2010). Recently, employment of background subtraction algorithms for moving object detection has not received enough attention. Despite a mature approach, the outcomes of such binary foreground regions provide limited information for the next tasks in the way of developing a complete and robust traffic analysis framework.

Another motion-based approach for detecting traffic vehicles is developed based on a fundamental computer vision method named Optical Flow (Lucas & Kanade, 1981). Theoretically, a moving object can be recognized via the optical-flow vectors extracted from the object region. These vectors are distinct from the background optical-flow vectors. This approach is generally employed for rejecting ego-motion of cameras (cam-shake) mounted on a moving observer (e.g. a vehicle) to detect other vehicles on roads (Baehring, Simon, Niehsen, & Stiller, 2005; Jazayeri, Cai, Zheng, & Tuceryan, 2011; K. Yamaguchi, Kato, & Ninomiya, 2006). Arrospeide et al. propose a method that includes coupling of optical flow computation for detecting hypothesized objects and an outlier rejection algorithm (Mahalanobis distance based classifier) for confirmation of the final results (Arróspeide, Salgado, Nieto, & Jaureguizar, 2008). Detecting traffic vehicles from an airborne drone is also investigated (Yu & Medioni, 2009). Since a vast region is captured on aerial images, the authors aim to track traffic flow before focusing on vehicle detection. To do so, only a limited area corresponding to roads is effectively analyzed to identify vehicles by means of the motion-pattern difference. Some drawbacks of optical flow based algorithms such as background movement, estimation of flow velocity, and computational overload are studied (Dessauer & Dua, 2010). Herein, an input image is processed by wavelet decomposition method before it is handled

with optical flow computation. A comparative study, in which vehicle detection is specifically subjected, among the three most popular optical flow methods is conducted (Głowacz, Mikrut, & Pawlik, 2012). These optical flow methods (Brox, Bruhn, Papenber, & Weickert, 2004; Horn & Schunck, 1981; Lucas & Kanade, 1981) are employed for detecting and counting traffic vehicles on a street intersection. The authors find out that the Horn-Schunck algorithm is the most effective for the aforementioned particular application. Occlusion challenge in vehicle detection is also addressed by implementing the optical flow and the active learning algorithm along with a calibrated stereo camera system (Ohn-Bar, Sivaraman, & Trivedi, 2013).

Recently, there has been a transition from utilizing motion-based methods to appearance-based approaches. This is due to the fact that the features extracted from motion-based algorithms provide inadequate information for the next tasks of vehicle understanding such as vehicle tracking, recognition, and classification. In addition, the features obtained from appearance-based methods allow detection and classification of vehicles from even a single image instead of a sequence of images. In the pool of apparent features employed for vehicle detection, some of them such as edge, symmetry, SIFT, Gabor etc., HOG and Haar-like features have been routinely exploited in the literature because of their high performance. The Histogram of Oriented Gradient (HOG) descriptor introduced by Dalal and Triggs comprises gradients and orientations of cells inside an image window (Dalal & Triggs, 2005). Following the implementation by Wijnhoven and De With, this feature is extracted from different poses of vehicles and then introduced to a simple linear classification for obtaining a detector (Wijnhoven & De With, 2011). Since variety of vehicle poses are utilized to train the classifier, the detector could successfully recognize vehicles under divergent views. Cheon et al. propose a way to estimate hypothesized vehicle positions using shadow detection, and then to extract HOG features from those hypothesized areas

in the image to confirm positive vehicles (Cheon, Lee, Yoon, & Park, 2012). Vehicles in dense urban areas are also identified from an aerial camera (Tuermer, Kurz, Reinartz, & Stilla, 2013). In this paper, the authors try to separate road regions from other city areas based on height information from the global Digital Elevation Map (DEM). Subsequently, HOG features are extracted from the separated road areas for recognizing vehicles. Using HOG features and the Latent-SVM (Support vector machines) on the Deformable Part Model (DPM), Felzenswalb et al. develop one of the best detection algorithms that has received an extensive attention in computer vision community recently (Felzenszwalb, Girshick, McAllester, & Ramanan, 2010). The DPM based algorithm that yields appealing results is specially applied for on-road vehicle detection (Niknejad, Takeuchi, Mita, & McAllester, 2012). Another comparative study among three active learning approaches compares the most popular apparent features, HOG and Haar-like, for vehicle detection (Sivaraman & Trivedi, 2014). The research provides a general observation in terms of time spent for annotating, amount of data required, recall, and precision of appearance-based approaches for vehicle detection.

Using Haar-like features is even more favored than utilizing HOG descriptor for vehicle detection due to two main reasons. First, Haar-like features are suitable for detecting the horizontal, vertical and symmetric structures such as vehicles. Second, these features are determined by applying the integral image method at very fast computation speeds, which grants real-time performance in practice. Inspired from the well-known Viola and Jones's face detection algorithm (Viola & Jones, 2001), one of the first Haar-like based study for vehicle detection is introduced by Ponsa et al. (Ponsa, López, Lumbreras, Serrat, & Graf, 2005). This research could detect vehicles by tracking 3D positions of those by following a sequence. The Haar-like features extracted from rear-view images of vehicles are delivered to a boost-classifier for obtaining hypothesized vehicle

regions. After that, a coupling between the perceived horizon and the actual width of the vehicles is implemented to estimate the 3D locations of the vehicles for verifying the hypotheses. Haselhoff et al. deploy a study to measure the influence of training image resolution on classification performance (Haselhoff, Schauland, & Kummert, 2008). The research shows that an optimization of image resolution could be learnt to mitigate computational load, which allows a faster performance. Sivaraman and Trivedi introduce a robust method for vehicle recognition and tracking (Sivaraman & Trivedi, 2010). Following this practice, vehicles could be recognized by using Haar features and a novel active-learning framework. Another study focusing on Haar-like features is proposed by Rios-Cabrera. In this research, a complete framework for detection, tracking, and matching vehicles through a tunnel is implemented with the aim of overcoming numerous real-life challenges such as bad illumination and poor image quality in tunnels (Rios-Cabrera, Tuytelaars, & Van Gool, 2012). An algorithm for vehicle detection and inter-vehicle distance estimation is proposed (G. Kim & Cho, 2012). First, Haar-like features of car-rear shadows are obtained to determine hypothesized vehicles. After that, directional edge features are employed to verify these hypotheses. The inter-vehicle distance is also calculated by means of estimating actual width of the detected vehicles and their locations. Park and Hwang introduce an improved descriptor named Haar Contrast Feature that could efficiently operate under various illumination conditions (Park & Hwang, 2014). The proposed descriptor compensates for diversified image contrast by a normalization factor, which is created from alteration of average intensity between consecutive image frames. The improved Haar features are verified on vehicle detection to confirm its outperformance under a wide range of illumination conditions in real-life environment.

Regarding the classification, most of the studies categorize vehicle types based on the information obtained from detected vehicles. For the cases where motion-based methods are used, classification parameters are generally dimensional features such as area, length, and height of bounding boxes (Gupte et al., 2002; Hsieh et al., 2006; Huang & Liao, 2004; Kanhere & Birchfield, 2008; Rad & Jamzad, 2005). Since these algorithms are mostly based on low-level image features, results are not really appealing and robust. Implementation of vehicle type classification by employing HOG features and SVM are provided by numerous papers (Z. Chen & Ellis, 2011; Z. Chen, Ellis, & Velastin, 2012; Khan, Cheng, Matthies, & Sawhney, 2010; Ng, Suandi, & Teoh, 2014). Moreover, some other authors even propose algorithms that could classify brand names and also logos of vehicles (Llorca, Arroyo, & Sotelo, 2013; B. Zhang, 2013). Due to Haar-like features providing less information than HOG descriptors for classification task, there are limited number of authors utilizing Haar-like features (Sam & Tian, 2012; Wang, Hsieh, Han, & Fan, 2014; B. Zhang, Zhou, & Pan, 2013).

Vehicle localization on the world coordinate system is seemed to receive less attention than vehicle detection and classification. Several studies introduce frameworks for determining vehicle locations in terms of intrinsic and extrinsic camera parameters (Levinson, Askeland, Dolson, & Thrun, 2011; Parra, Sotelo, Llorca, & Ocaña, 2010; Sivaraman & Trivedi, 2013). Following these papers, a mathematical function is developed from the parameters that are derived by the calibration of the camera, representing relationship between the image coordinates and the world coordinates. Once the function is established, the world locations of vehicles could be obtained from their image positions pinpointed by the detected bounding boxes. Another approach for locating vehicles based on stereo technique is conducted in some studies (Senlet & Elgammal, 2011; Sivaraman & Trivedi, 2011; Wu & Ranganathan, 2013). Stereo method is a mature

technology in computer vision to recover 3-D objects from two 2-D images captured by two synchronized cameras. Since the pixel disparity of a certain point on both left and right cameras can be measured, a distance from that point to cameras is resolved for archiving its 3-D coordinates. Although both mentioned approaches could localize vehicles, camera calibration can be complicated and challenging, especially for in field deployment.

Motivation and Objectives

It is seen that vision based vehicle detection protocols have been immensely developing in the last decade. These studies are categorized into two groups named motion-based and appearance-based methods. Based on numerous fundamental algorithms in computer vision such as background subtraction, frame differencing, and optical flow, motion-based techniques seem to attract more interest at the beginning; however, some noted shortcomings have been pointed out later on. Due to fast evolution of high-level imaging features, such as symmetry, SIFT, Gabor, HOG, Haar-like, etc., vehicle detection algorithms are recently switching to the appearance-based group, especially using outperforming HOG and Haar-like approaches. While HOG features are obtained by calculating pixel intensity gradients and orientations of cells inside an image window, Haar-like features are determined by convolving Haar wavelets kernels. The features extracted either by HOG or Haar-like approaches are transmitted to training programs such SVM and/or Boost classifiers for obtaining detectors. The trained detectors are then employed on an image of highway (road) traffic for distinguishing hypothesized vehicles. Although it is difficult to conclude which feature (HOG or Haar-like) surpasses another, Haar-like based techniques are more efficient in computation because these features are calculated by means of the integral image execution.

Although using Haar-like methods for detecting vehicles is very fast and potent, employing these features yields rather global information than local aspects of a particular vehicle type. Thus, conducting HOG descriptors for classifying vehicle types seem to attract more interest in the literature. Using dimensional features obtained from motion-based approaches has also been handled for vehicle type categorization. However, the features such as area, length, and height attained from bounding boxes of detected vehicles bring inadequate inputs for developing a robust vehicle type classifier. Regarding the vehicle localization, two fundamental techniques are widely utilized including stereo and camera model. Despite these methods are able to recover 3-D positions of vehicles on the world coordinate system, complex camera calibration is a challenge as for real-life implementation.

In this Chapter, a selection of suitable computer vision algorithms is implemented to serve towards the research aim, which is to obtain traffic information including weights of vehicles and their positions while passing over a bridge deck. The objectives are comprised of two modules: 1) vehicle categorization by detection of vehicle types, and 2) localization of detected vehicles on the bridge deck defined by the world coordinate system. Based on the investigation conducted in the previous sections, HOG descriptors will be selected in service of the first module. The findings of the first module such as bounding boxes are provided as inputs for the second module, that is, vehicle localization. Since it is known that a bridge deck can be posited as a plane, a proper plane geometric transformation technique is proposed for calculating positions of vehicles by means of world coordinates on the bridge deck. The methodology details of the protocol are presented subsequently. After that, the efficiency of the procedure is verified through multiple tests conducted in the UCF Structural Laboratory on a small-scale bridge.

Theoretical Background

The flowchart illustrating the implementation for detection and localization of traffic vehicles passing over a bridge is seen in Figure 32. In brief, a surveillance camera is mounted at a high position overlooking the whole deck of the bridge. The images acquired by the surveillance camera are analyzed to detect every type of vehicles that may appear on the bridge deck. The detected vehicles are then labeled by bounding boxes matching with their types. Finally, the positions of bounding boxes on the 2-D image coordinates corresponding to detected vehicles are transformed to the 2-D bridge deck coordinates based on the plane geometry transformation.

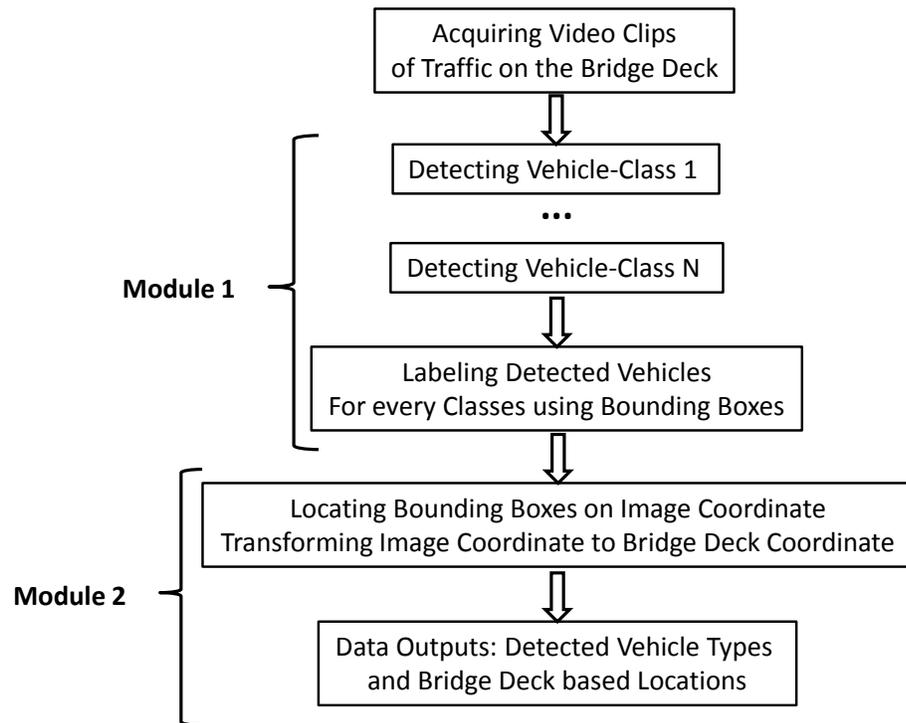


Figure 32. Protocol for detection and localization of traffic vehicles passing over a bridge based on computer vision approach

Imaging Features: Histograms of Oriented Gradients (HOG) Descriptors

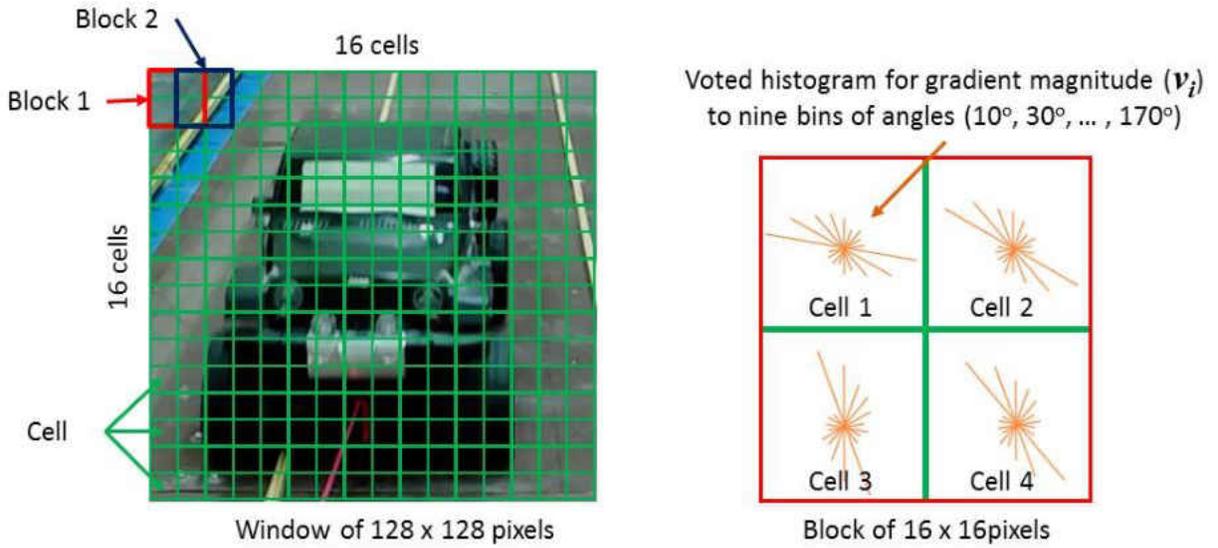
Histograms of Oriented Gradients (HOG) descriptors are introduced as a sort of robust features for human detection (Dalal & Triggs, 2005). In that study, the HOG based human detection significantly outperforms the other human detection techniques upon a certain time. From then, the HOG descriptors have been successfully being utilized for detecting other types of objects including vehicles. As suggested in the name itself, Histograms of Oriented Gradients (HOG) descriptor is a vector comprising intensity gradient orientations and magnitudes of cells inside an image window (so-called patch).

To determine an HOG descriptor of an image window $W(x,y)$, the intensity gradient magnitudes and orientations of the window W are required. The equations leading to the derivation of descriptor are as follows:

$$m(x, y) = \sqrt{(W(x + 1, y) - W(x - 1, y))^2 + (W(x, y + 1) - W(x, y - 1))^2} \quad (28)$$

$$\theta(x, y) = \tan^{-1} \left(\frac{W(x, y + 1) - W(x, y - 1)}{W(x + 1, y) - W(x - 1, y)} \right) \quad (29)$$

where $m(x,y)$ is the gradient magnitude, and $\theta(x,y)$ is the orientation at a pixel location (x,y) in the image window W . Subsequently, the image window W is divided into cells, for instance, 16 x 16 cells. A block is developed from 4 neighboring cells. Each block is then placed on each other by 50% overlap across the image window. For an image window defined by 128 x 128 pixels, the number of blocks yield to be 15 x 15 = 225. As the cell size is of 8 x 8 pixels, each block has its size of 8 x 8 pixels. The details of separation on the given 128 x 128 pixels image window are illustrated in Figure 33. Apparently, other division scenarios can be conducted for a particular case; however, performances of the descriptors corresponding to these scenarios are not greatly changed (Dalal & Triggs, 2005).



$$\text{Concatenated HOG vector: } H = \{v_1, v_2, \dots, v_{n-1}, v_n\}$$

Figure 33. Protocol for determining HOG features; example for a given 128 x 128 pixels image window with cell size of 8 x 8 pixels and block size of 4 x 4 cells

Since the cell size is of 8 x 8 pixels, there are sixty four (64) values of gradient magnitudes as well as another sixty four (64) values of gradient orientations, which are calculated from Equation 28 and 29. These (64) values of gradient orientations are quantized into nine (9) bins such as 10°, 30°, 50°, ... , 170°, and the vote for each orientation is its gradient magnitude. After quantizing, a cell information is presented by nine (9) values of summarized-vote magnitudes corresponding to the (9) bins of angles. As the HOG descriptor vector of an image window $W(x,y)$ is constructed by concatenated information of every cell, number of elements in the vector yields as illustrated in Equation 30.

$$n = \text{blocks} \times \text{cells} \times \text{bins} = 225 \times 4 \times 9$$

$$n = 8,100 \text{ (elements)}$$
(30)

Let H be a certain HOG vector, then it is represented as below.

$$H = \{v_1, v_2, \dots, v_{n-1}, v_n\} \quad (31)$$

where v is the voted histogram value; and n is number of histograms (e.g. 8,100 elements) calculated from Equation 30. Some examples for HOG extraction on various objects such as a sedan car, a bike, and a bottle are shown in Figure 34. Herein, the H vectors are depicted as imaging fashion to observe the shapes of objects having been reflected by HOG descriptors.

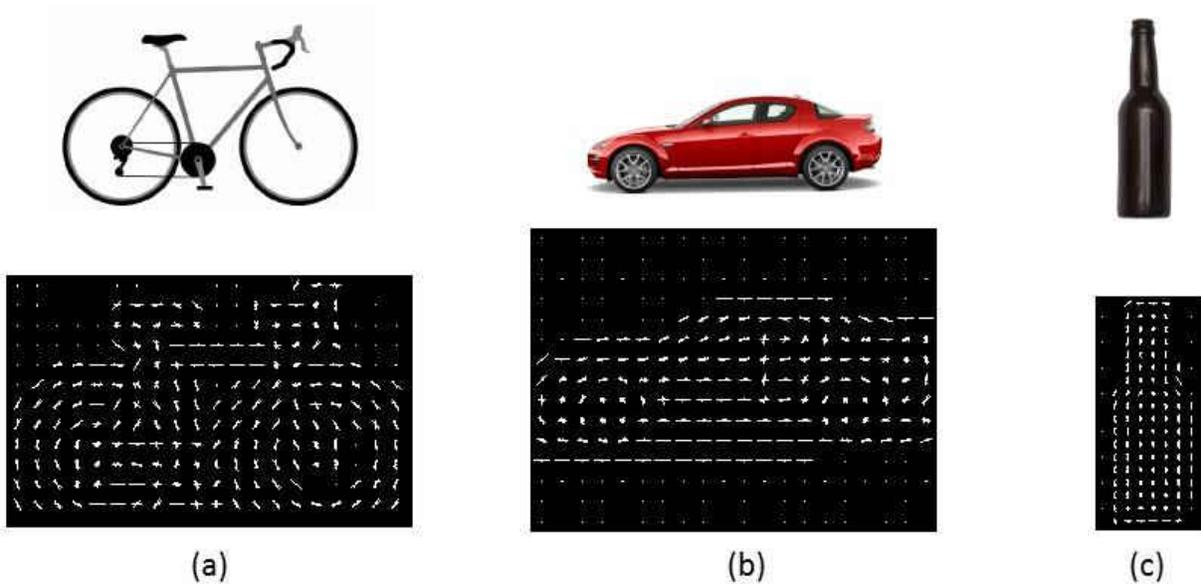


Figure 34. HOG feature extraction represented as imaging fashion for example (a) a bike, (b) a sedan, and (c) a bottle

Detection of Vehicle Types based on AdaBoost Technique and Cascade Classifier with HOG Features

Detecting objects from images is one of the most interesting challenges in computer vision. Generally, a detector is developed to detect a particular type of object such as human, conveyance means (bikes, vehicles, planes, boats, etc.), animal (cats, dogs, birds, horses etc.), and other miscellaneous. Mathematically, a detector comprises complex mathematical functions that can discriminate and then recognize a certain object from a scene based on the object features. Despite

there are several approaches to develop various sorts of detectors, using machine learning techniques to train a detector is the most popular way due to its efficiency and robustness. As mentioned previously, the HOG features are utilized for developing detectors in this dissertation research. Since there are many elements in an HOG vector (following Equation 30), dealing with such huge number of elements is prohibitively expensive for any classification algorithms. Thus, only a limited number of features with outperformance are selected. After that, detectors are built up on these crucial features. Herein, AdaBoost technique is appointed for two purposes being selection of crucial features and development of vehicle-detectors. The approach is acknowledged as a very powerful tool and has been also conducted in numerous object detection studies.

Introduced by Freund and Schapire, Adaboost is a very popular boosting technique that combines poor performance classifiers (so-called weak classifiers) into a complex one with much higher achievement (Freund & Schapire, 1997). In this context, a weak classifier can be any “rule of thumb” guessing that performs better than random (50 percentage of chance). A weak classifier can be developed by means of feature distributions that are evolved from training of data sets. For example, based on the reliable statistical datasets, distributions of human height (a sort of human feature) in a city can be constructed for males and females separately denoting as the male-height and the female-height distributions. By setting a height threshold on these distributions, a weak classifier can be attained to categorize males and females with an accuracy greater than 50%.. Let $f_i(i)$ be the i^{th} weak classifier for identifying a particular type of object on *image* (i). Since $f_i(i)$ is a rule of thumb guessing that yields to zero (0) or one (1) for a negative or positive object respectively, the boosted classifier for *image* (i) becomes:

$$F(i) = \text{sign} \left(\sum_{t=1}^T \alpha_t f_t(i) \right) \quad (32)$$

where $F(i)$ is the final (strong) classifier that also yields zero (0) for a negative object and one (1) for a positive object; T is the number of available weak classifiers; and α_t is the weight corresponding to the t^{th} weak classifier. These weight values are the key parameters of the AdaBoost technique assigned by the training process that emulate error rate of the weak classifiers. For instance, a weak classifier with 50% accuracy will be assigned a weight value of zero (no benefit for the classification) whereas a good performance classifier has its positive weight but a bad performance one is assigned a negative value of weight. By ranking the performance of many better-random classifiers and putting them into construction of the final classifier through certain weighing values, the final classifier is obtained to provide a much higher achievement of categorization results. The details of the AdaBoost technique involving its mathematical equations and processes can be seen in the related reference (Freund & Schapire, 1997).

Implementation of the AdaBoost technique for selecting outperforming features is firstly introduced by Tieu and Viola (Tieu & Viola, 2000). Following that paper, features as elements in a vector $H = \{v_1, v_2, \dots, v_{n-1}, v_n\}$ (Equation 31) can be ranked by developing a strong classifier from various amount of weak classifiers, each of which is developed from a single feature. In other words, for a particular feature v_t , a weak classifier $f_t(i)$ is established as follows.

$$f_t(i) = \begin{cases} 1, & \text{if } v_t(i) < \theta_t \\ 0, & \text{otherwise} \end{cases} \quad (33)$$

where θ_t is a threshold appointed for the v_t feature; and t is from 1 to n . Since there are n number of weak classifiers, the strong classifier is able to be obtained by utilizing the AdaBoost technique. The form of the final classifier derived from Equation 32 will be:

$$F(i) = \begin{cases} 1, & \text{if } \sum_{t=1}^n \alpha_t f_t(i) \geq \frac{1}{2} \sum_{t=1}^n \alpha_t \\ 0, & \text{otherwise} \end{cases} \quad (34)$$

where α_t is the weight of the t^{th} weak classifier corresponding to the v_t feature. Although this process yields a compelling classifier, it results in idle computation due to the usage of whole set features (Viola & Jones, 2001). Here, the strength of the method comes from the set of weight α_t . By ranking these weight values, the crucial features are recognized with higher weights for the purpose of developing an even more powerful classifier in discriminating and with less computation time. The protocol using the crucial features for creating a strong classifier is subsequently described as follows.

Since using a huge set of features for developing a classifier causes ineffective computation in return, Viola and Jones construct a cascade of classifiers that bring more robust detectors while radically mitigating computation time (Viola & Jones, 2001). The conceptual insight is that *the less the number of features used, the faster the ciphering classifier is*. The protocol of the cascade implementation is depicted in Figure 35.

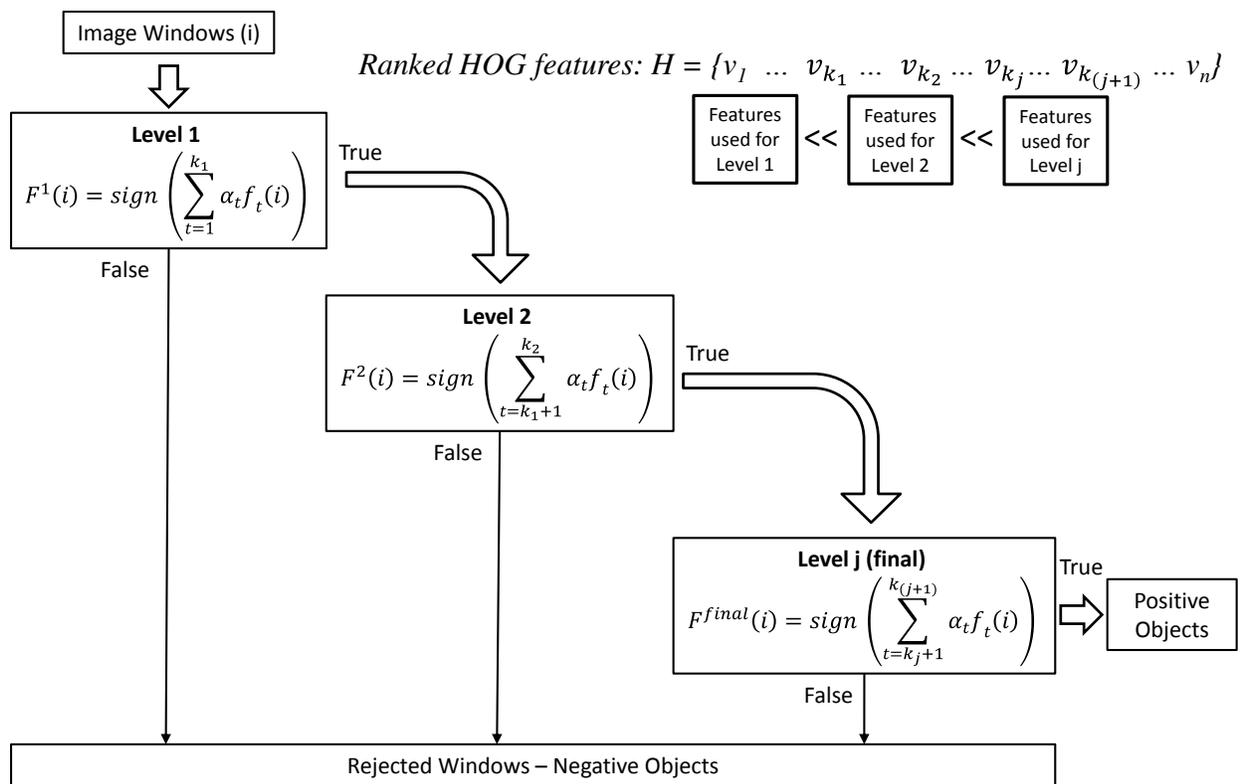


Figure 35. Schematic for a cascade of classifiers. Each level is a strong classifier that boosted from many weak classifiers

It is seen that the cascade comprises finite levels of boosted classifiers. At each level, a very limited number of features is utilized for establishing a boosted classifier. For instance, the obtained crucial features are concatenated starting from the best going down to the worst performance. Then, only the strongest feature is used for the first level boosted classifier, the next five (5) features for the second level, the subsequent ten (10) features for the third level, and so on. The figure also shows that the positive objects from a certain level (including true positive and false positive objects) will be bestowed to the next level, which is developed by using more features. Hence, a less feature classifier deals with more inputs (such as image windows) while a more feature classifier handles much less inputs, which is achieved by the current computation efficiency. Moreover, since a more feature classifier (more robust classifier) deals with a few

inputs that have high rate of positive objects, the detection rate of the whole cascade immensely increases.

A detector is obtained by training *the AdaBoost based cascade classifier* with labeled image datasets of positive images and negative images respectively. For instance, a truck-detector is developed by means of learning from a set of truck images that are called as positive images followed by negative images which can be another set of non-truck images such as landscape images or other types of vehicle (sedan, bus, etc.) images. In this Chapter, an off-the-shelf *AdaBoost based cascade classifier* embedded in MatLab is used for obtaining detectors corresponding to each vehicle classes. Obviously, the training image datasets (positive and negative image sets) must be manually prepared by the user. The training process is illustrated in Figure 36.

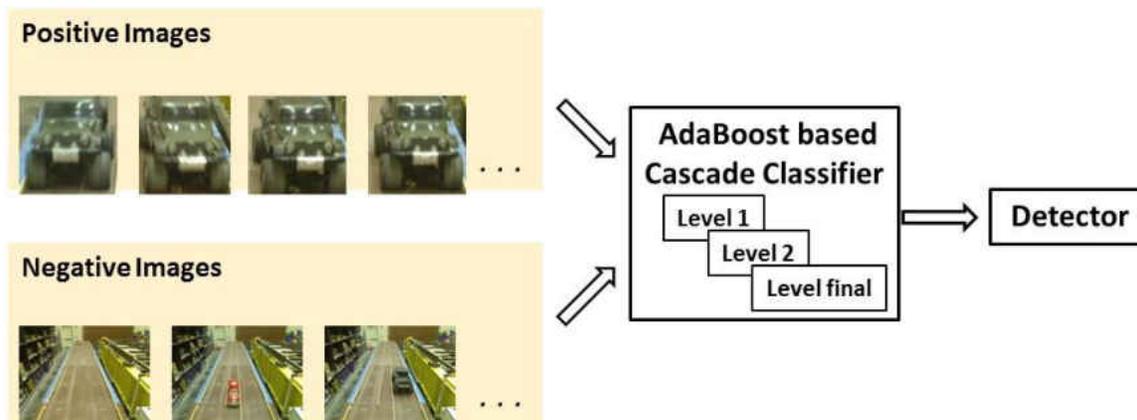


Figure 36. The training process for obtaining a detector

Once the detector is obtained, it is utilized for identifying the objects by a scanning window at any region on an image. At a particular region, the scanning window crops the input image getting an image window. Then, the image window is fed to the detector developed by *AdaBoost based cascade classifier* to find out if the constituents of the window is either object or non-object.

Since object sizes are varied, scanning window dimensions must be scaled from the smallest up to the biggest possible ratios to detect all available objects on the input image. If there are multiple detectors for different types of objects, the procedure will be repeated for each detector one at a time. The detection scheme is described in Figure 37.

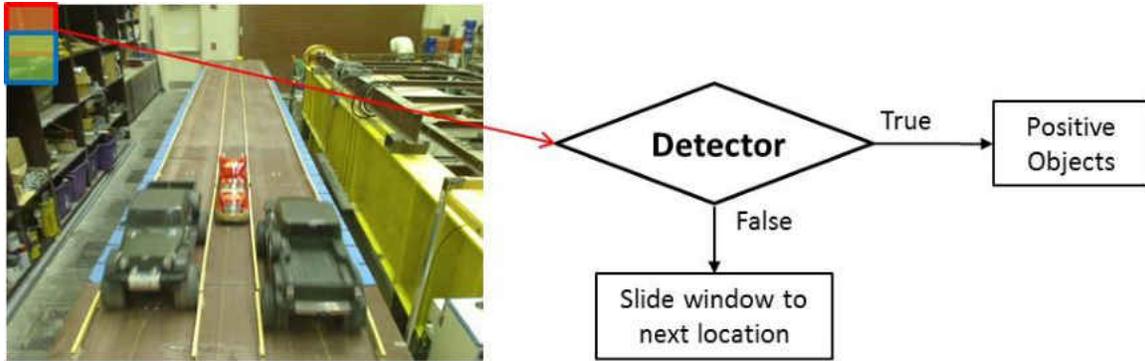


Figure 37. The detection scheme using a trained detector

Verification of the detection framework for vehicle types based on AdaBoost technique and cascade classifier with HOG features is conducted on subsequent section. The results, shortcomings, and how to overcome false detection are also discussed therein.

Localizing Detected Vehicles based on Geometry Transform Approach

As it is mentioned at the beginning of this Chapter, determining the positions of loads on structures is as essential as attaining their amplitudes. This is due to the fact that altering load locations on a structure might come off in forms of extremely fluctuated magnitudes of responses. Since vehicles can be detected and then matched with predefined catalog of vehicle classes, weights of these vehicles are assigned to the mean weight of corresponding classes. Furthermore, the detected vehicles are also localized on the 2-D image coordinates by bounding boxes, which are appointed by the detector (Figure 38).

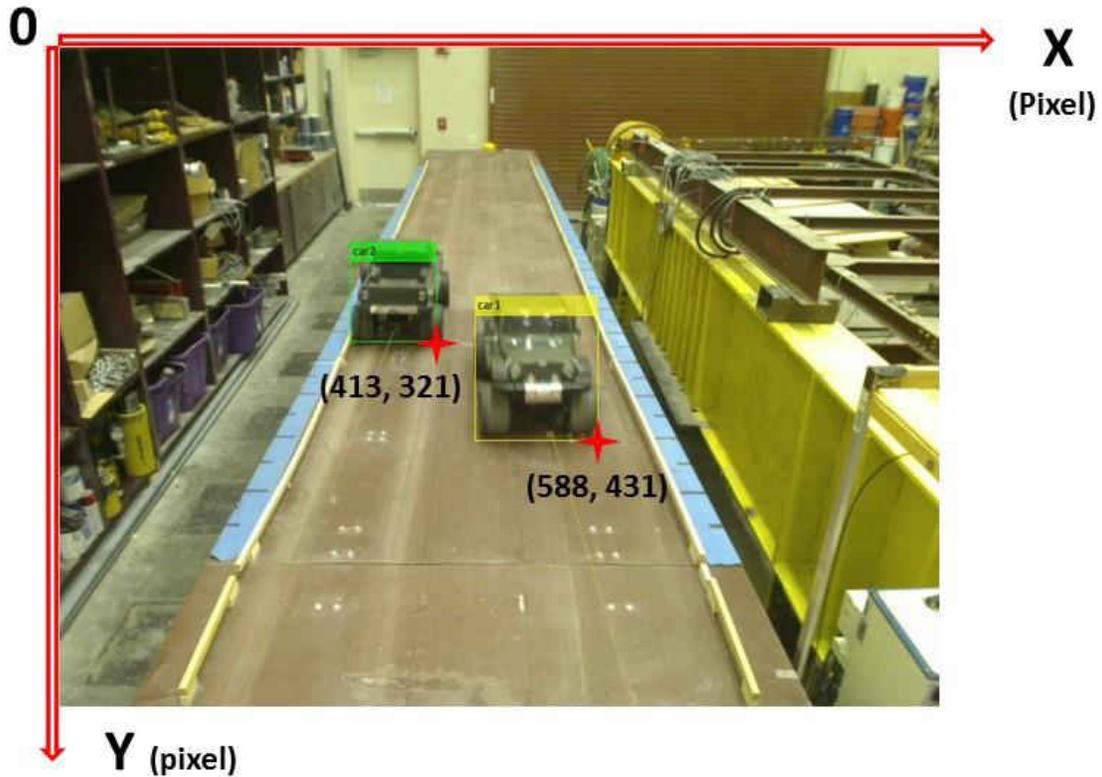


Figure 38. Two detected vehicles are located on the 2-D image coordinates by bounding boxes; for example, the bottom-right corners of boxes are obtained in the pixel-unit

In this study, only locations of vehicles in the 3-D world coordinate system are of interest rather than those on 2-D image. Hence, this section aims to convert points on a 2-D image to 3-D world points belonging the Cartesian coordinate system predefined on the bridge deck. From literature review, camera calibration is an imperative technique in 3-D computer vision for obtaining metric information from 2-D image. Although many studies have been conducted, most of them are developed in terms of a fundamental equation for camera model. Let a 2-D point on an image be denoted by $p = (u, v)^T$; and the corresponding point in 3-D world coordinate system by $P = (X, Y, Z)^T$. Then camera model equation is written as:

$$p = [M] [E] P \quad (35)$$

The M matrix is the camera intrinsic matrix to link the pixel coordinates of an image to the corresponding metric coordinates in the camera sensor. This matrix comprises parameters such as perspective projection (focal length) and some scale factors to transform between pixel coordinates to camera sensor coordinates. It is presented as below

$$M = \begin{bmatrix} -fk_x & 0 & u_0 & 0 \\ 0 & -fk_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (36)$$

where f is the focal length of the camera; k_x and k_y are the effective size of pixel in mm; and (u_0, v_0) is the coordinates of the principal point.

The matrix E in Equation 35 is the camera extrinsic matrix. Since it is actually a rototranslation matrix for transforming between the image sensor coordinates and the predefined 3-D world Cartesian coordinate system, a 3 by 3 rotation matrix and a 3-D translation vector are included

$$E = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & t_x \\ r_{2,1} & r_{2,2} & r_{2,3} & t_y \\ r_{3,1} & r_{3,2} & r_{3,3} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (37)$$

where $r_{i,j}$ is the coefficient of the rotation matrix; and t_x , t_y , and t_z are the distances between the origin of the camera sensor coordinate system and the origin of the 3-D world coordinate system in x -axis, y -axis, and z -axis, respectively.

Thus, the Equation 35 takes the form of

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} -fk_x & 0 & u_0 & 0 \\ 0 & -fk_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & t_x \\ r_{2,1} & r_{2,2} & r_{2,3} & t_y \\ r_{3,1} & r_{3,2} & r_{3,3} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (38)$$

To form the relationship depicted in Equation 38, all parameters in the camera extrinsic and intrinsic matrices must be obtained. Although this challenge could be overcome by some basic camera calibration methods, the implementations require procedures that would be difficult to conduct in the field for real-life structures (Sturm & Maybank, 1999; Triggs, 1998; Z. Zhang, 2000).

Fortunately, it is seen that the vehicles travel on a bridge deck that is mostly planar. Since the vehicle loads of interest act vertically on bridge structures, elevations of those insignificantly alter responses of the structures acquired by sensors. This observation enables an alternative protocol to calculate the positions of vehicles on the 2-D bridge deck instead of in 3-D space. The assumption of planar bridge deck is also made in a previous research (R. Zaurin & Catbas, 2010). In that paper, Zaurin and Catbas adopt this assumption for determining distances from a vehicle to a camera in terms of Lagrange Interpolation method.

In consideration of planar bridge deck, a point on the bridge deck is denoted by $P = (X, Y)^T$ as $Z = 0$. Therefore, the Equation 38 becomes

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} -fk_x & 0 & u_0 \\ 0 & -fk_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{1,1} & r_{1,2} & t_x \\ r_{2,1} & r_{2,2} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (39)$$

Or

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (40)$$

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = [T] \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

Where T is named as transformation matrix mapping the coordinates from the image to the real-life coordinates set on the bridge deck. From the derivation of Equation 40, we have

$$\begin{aligned}
u &= \frac{a_1X + a_2Y + a_3}{a_7X + a_8Y + 1} \\
v &= \frac{a_4X + a_5Y + a_6}{a_7X + a_8Y + 1}
\end{aligned}
\tag{41}$$

And then

$$\begin{aligned}
a_1X + a_2Y + a_3 - a_7uX - a_8uY &= u \\
a_4X + a_5Y + a_6 - a_7vX - a_8vY &= v
\end{aligned}
\tag{42}$$

The Equation 42 is rewritten in a matrix form

$$\begin{bmatrix} X & Y & 1 & 0 & 0 & 0 & -Xu & -Yu \\ 0 & 0 & 0 & X & Y & 1 & -Xv & -Yv \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{bmatrix} = \begin{bmatrix} u \\ v \end{bmatrix}
\tag{43}$$

These coefficients, a_1 to a_8 , are calculated by assigning a set of reference points on the bridge deck. The coordinates of reference points in the image coordinate system are denoted as (u_i, v_i) while the coordinates in the bridge deck are (X_i, Y_i) . Given n reference points, the total number of functions derived from Equation 43 is $2*n$. Eventually, the coefficients are calculated by means of the Least Square Fit algorithm. The insight of plane transformation is illustrated in Figure 39, while the details and numerous implementations are described on the next section.

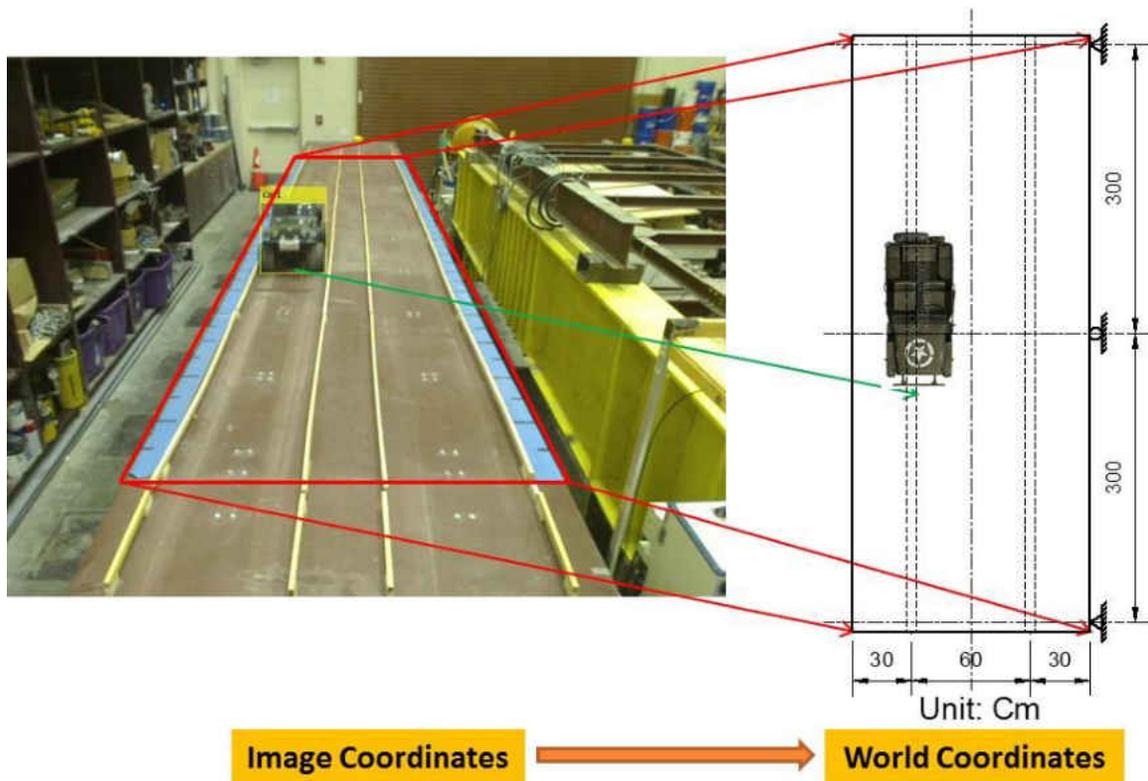


Figure 39. Geometry transformation between the 2-D image and the 2-D bridge deck coordinates

Laboratory Verification

Experiment Design: UCF 4-Span Bridge, Camera, and Vehicle Classes

The verification for detection of vehicle type framework, which is expressed on the “Theoretical Background” section, is conducted at the Structural Lab of UCF. To simulate traffic passing over a bridge, multiple small-size cars are driven back and forth on the deck of the UCF 4-Span Bridge. As it is described on Chapter 2, that apparatus bridge consists of two 300cm main continuous spans and two 120cm approach spans. The bridge deck includes a 3.18mm steel sheet at 120cm wide which turns out the deck dimension of 600 x 120 cm to length and width respectively. To view the whole bridge deck, a camera is mounted on a steel pole placed at the end

of the bridge. The camera utilized herein is a Logitech Webcam C930e that can directly connect to a computer via a USB port. This webcam can capture full 1080p (1080 x 1920 pixels) High Density (HD) video clips at a speed of 30 frames per second (30 Fps). Since other cameras have also been employed, it is seen that there is no requirement on any type of cameras. The experimental setup is shown in Figure 40.

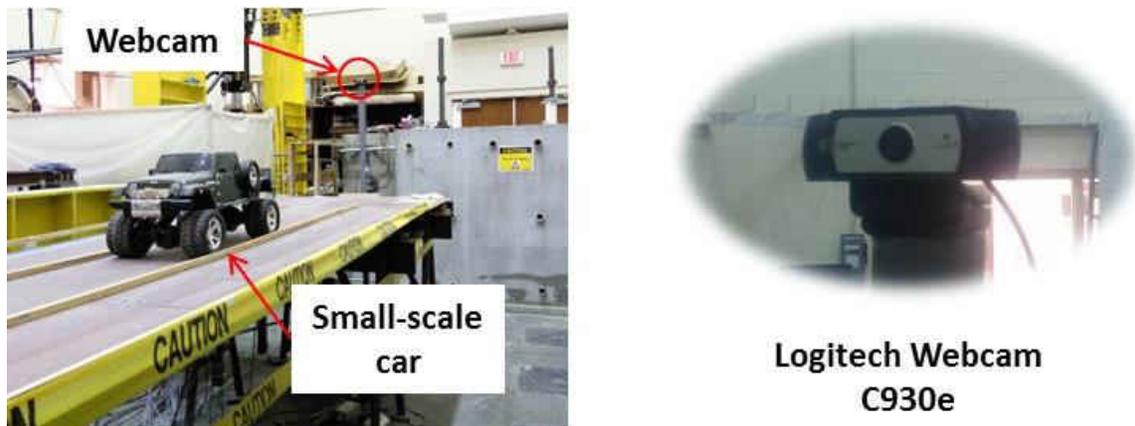


Figure 40. The UCF 4-Span Bridge and experiment setup

Taking the bridge deck dimension (600 x 120 cm) into account, several small-scale cars are used. Those cars are divided into three classes named *Class 1*, *Class 2*, and *Class 3* with their own descriptions by their appearances, weight of axles, axle distances, and horizontal wheel distances. The specifications of each class are depicted in Figure 41. The tests are simply deployed by driving those small-scale vehicles back and forth on the 4-Span Bridge deck, while the camera is recording. Numerous driving scenarios are implemented by changing vehicle configurations. Those configurations comprise changing number of deployed vehicles as well as number of deployed classes, driving vehicles within the specified lanes or freely (no lane), and altering speed of vehicles (fast or slow). In total, forty (40) tests are carried out to cover those mentioned

configurations. Finally, the clips are analyzed using the proposed framework. Details of results including problems and challenges are discussed in subsequent section.

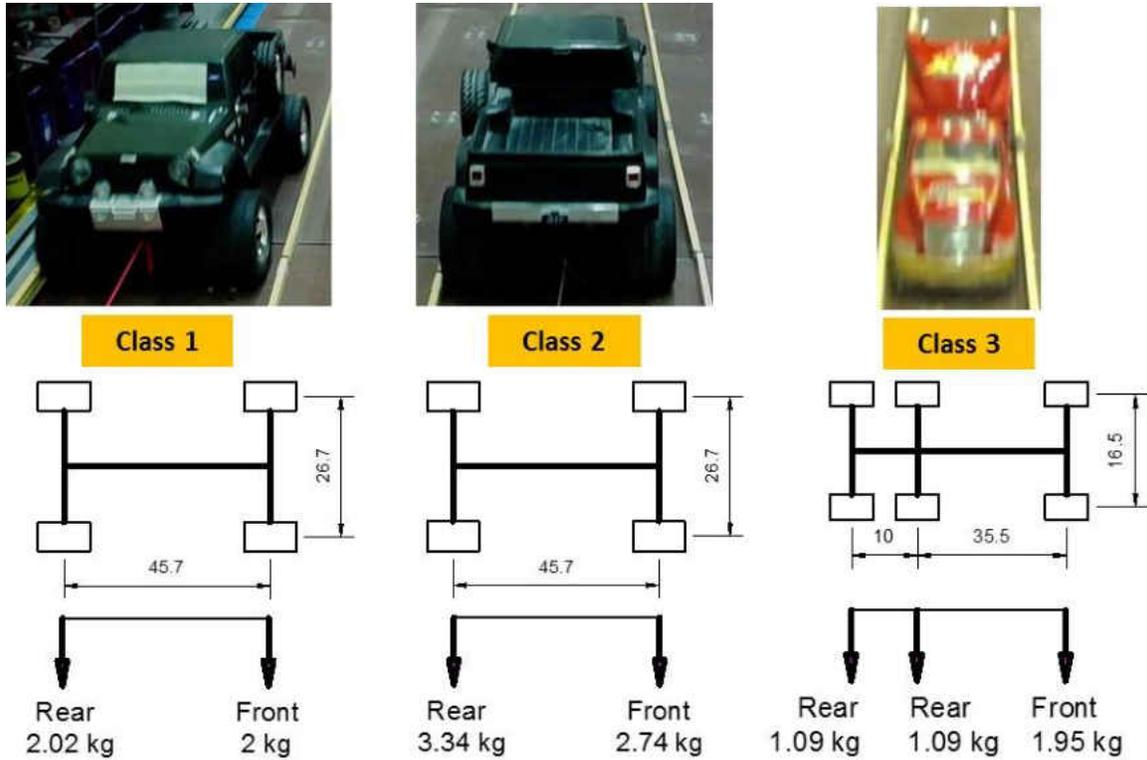


Figure 41. Vehicle classes and their specifications

Training Detectors using AdaBoost and Cascade Classifiers

As there are three vehicle classes, three vehicle detectors are developed namely *Detector 1*, *Detector 2*, and *Detector 3* corresponding to these classes in Figure 41. The procedure to train the detectors is explained in Figure 36. First, a set of positive images is prepared. For example, a total number of 102 images of the *Class 1* are labeled manually by cropping image windows of the *Class 1* vehicles from testing images. Those images for cropping are also subjectively selected by the engineer to make positive images covering all facets of a particular *vehicle Class* such as image window sizes of vehicle class (image scale) and views of vehicle poses. Although selecting

and cropping are manually performed, labeling is accomplished by using the *Image Labeler Apps* from MatLab. Meanwhile, negative images are picked out from non-vehicle images as well as images of *Class 2* and *Class 3* vehicles, for example, when developing *Detector 1*. It is affirmed that to select other *Classes* of vehicle images as negative ones for training a certain *Detector* is immensely important to obtain the discriminative and robust *Detector*. Preparing a set of negative windows is very simple when the training program sequentially crops windows from the selected negative images. As indicated previously, an off-the-shelf *AdaBoost based cascade classifier* embedded in MatLab is utilized for training to obtain *Detectors* in the end.

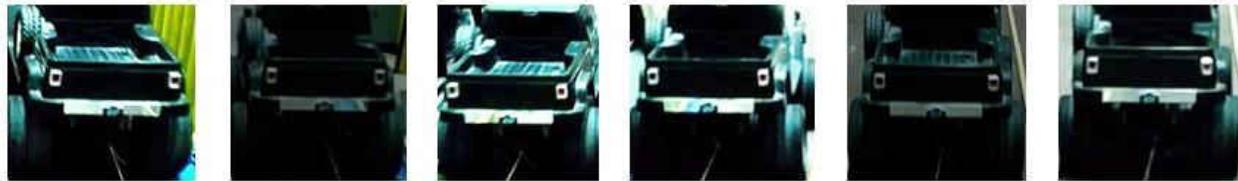
In summary, there is a total number of 102 positive images for training *Detector 1*, 104 positive images for training *Detector 2*, and 102 positive images for training *Detector 3*. The negative window sets are replicated from a pool of 20 negative images (1920 x 1080 pixels) for each class. To simulate the variation of illumination conditions, those positive and negative images are also modified in terms of brightness and contrast parameters. The number of cascades used for the training procedure are appointed as 10 cascades. Each *Class* of vehicle is trained independently to obtain its *Detector* such as *Detector 1*, *Detector 2*, and *Detector 3*, which is used for recognizing vehicle *Class* from an input image. The detection vehicle results are shown in the next section. Examples of positive and negative images that are used for training *Detectors* are illustrated in Figure 42.



Positive Images for Training Class 1



Negative Images for Training Class 1



Positive Images for Training Class 2



Negative Images for Training Class 2



Positive Images for Training Class 3



Negative Images for Training Class 3

Figure 42. Examples of positive and negative images for training classes (1,2,3) of vehicles

Class Vehicle Detection Results, and False Detection

Each *Detector* developed from the training module is consecutively used to scan across an input image at multiple scales and locations to find out the corresponding vehicle classes. The aim for using scalable windows is that vehicle image at any size appeared on the input image would be possibly identified (e.g. a vehicle that is further from the camera has a smaller size in the input image, and vice versa). The locations of the *Detectors* are obtained by shifting those scalable windows for some number of pixels. Selecting small values of scale for windows as well as shifting pixels yield more accurate creation of bounding-boxes, but tedious computation in return. Due to the fact that dimension and location of bounding boxes are both essential for the next vehicle localization task, the detection parameters are selected at small rates values, for example the *scale ratio of 1.05*, and the *shifting pixel of one* (1 pixel).

The results of vehicle detection for several different test scenarios such as single or multiple vehicles appearing on the deck are presented in Figure 43. It is seen that the detected vehicles are labeled by bounding boxes that have yellow, green, and red colors corresponding to the *Class 1*, *Class 2*, and *Class 3* respectively. Although vehicles can successfully be identified in most cases, some false detections are also recognized and depicted in Figure 44. The types of false detection encountered are undetectable vehicles, detected background regions, wrong type of class, and inaccurate dimensions of bounding boxes.

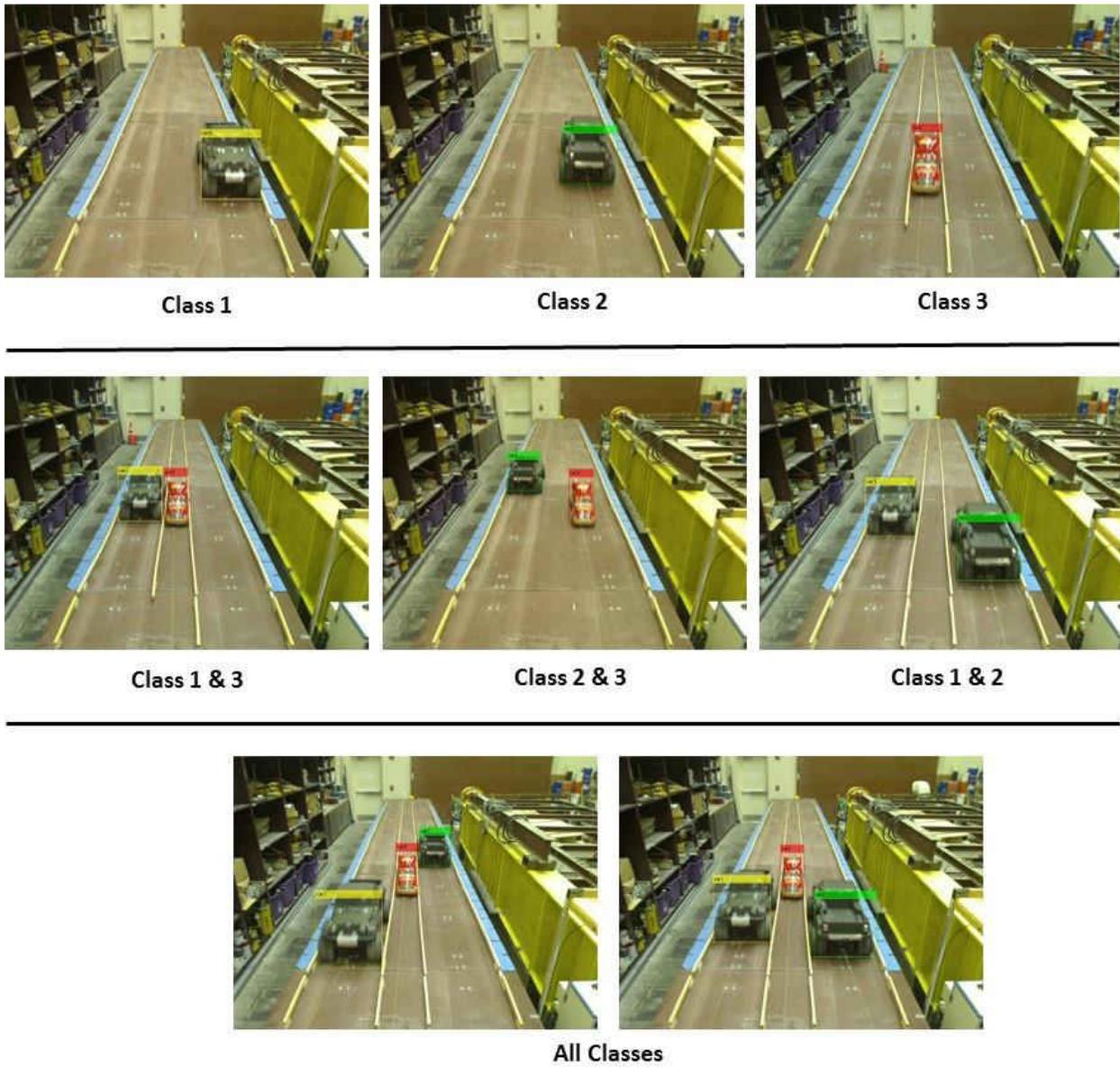


Figure 43. True positive of class vehicle detection

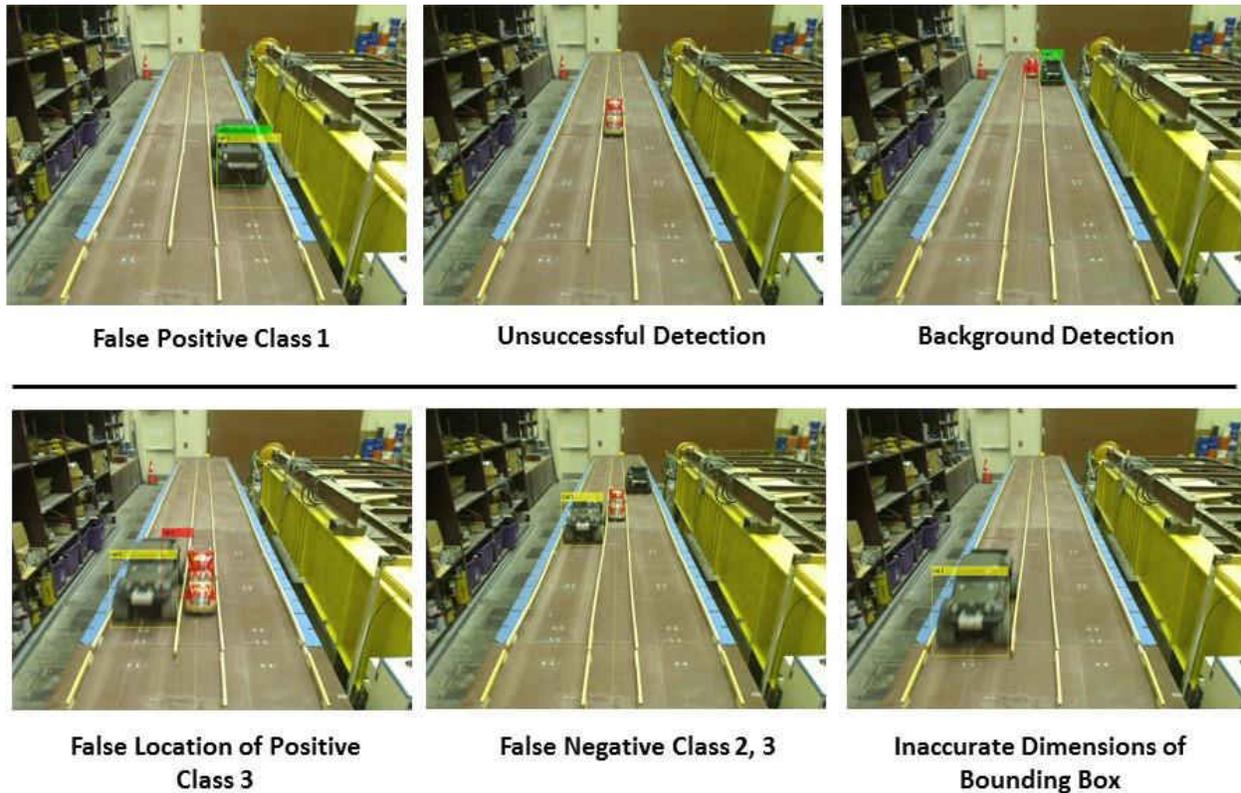


Figure 44. Examples of False Detection

The efficiency of detection procedure is evaluated based on the number of false detections out of 5826 images that are captured from forty tests. Since there is no ground-truth image database for verification, the accuracy of all 5826 images is visually verified by the researcher. The evaluation is conducted by comparing detection rates among three classes (Table 7) as well as between single vehicle and multiple vehicle scenarios (Table 8). Table 7 shows that the detection rate of the *Class 1* (95.9 %) is the best while the one of the *Class 2* (88.9 %) is the worst. That observation can be explained with front part of the *Class 1* vehicle having more textures that makes discrimination easier for classifier. During the evaluation process, it is also realized that most of the false detections occur when vehicles are far away from the webcam. This is due to the fact that vehicles driven away from the webcam commonly become smaller sized and blurry (textureless)

objects in images. Because of this shortcoming, using multiple cameras for vehicle tracking (Javed, Rasheed, Shafique, & Shah, 2003) is highly recommended for real life implementation especially for long bridges so that each camera could only focus on a particular section of a bridge (e.g. a span). Obviously, the detection rate for the case of single vehicle (92.5 %) is better than the rate when multiple vehicles passing over the bridge (87.2 %). In summary, the detection rates for all scenarios vary in a range from 87.2 % to 95.9 %. False detection problems are addressed and discarded in Chapter 4.

Table 7: Detection rates for each classes of vehicles

Vehicle Class	Class 1	Class 2	Class 3
Detection Rates	95.9 %	88.9 %	94.4 %

Table 8: Comparison of detection rates between cases of single vehicle and multiple vehicles on bridge deck

Number of Vehicles	Single	Multiple
Detection Rates	92.5 %	87.2 %

Vehicle Localization Results

As specified in the theoretical section, positions of detected vehicles are determined by transforming 2-D image coordinates of bounding boxes (Figure 38) to 2-D coordinates on the bridge deck coordinate system. This implementation can be achieved by developing the transformation matrix T , which is denoted in the following equation.

$$T = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & 1 \end{bmatrix} \quad (44)$$

The coefficients a_1 to a_8 of the matrix T are calculated by utilizing the Least Square Fit algorithm for relationship equations (each created by Equation 43) between a set of reference points on the bridge deck (X_i, Y_i) and its image (u_i, v_i) on the pixel-unit coordinates. The 2-D world Cartesian coordinate system on the UCF 4-Span Bridge deck is defined as in Figure 45. It is seen that there are fifteen reference points marked on the bridge deck by using LEDs (Light-Emitting Diode). Subsequently, those LEDs are also identified on the 2-D image coordinate system. The coordinates of reference points on the bridge deck (X_i, Y_i) and the image (u_i, v_i) are presented on Table 9. Please note that placing any sorts of references on a bridge deck is commonly simple, and it does not require any equipment. For a real-life structure, reference objects may be traffic cones or other similar things so that these could be seen clearly by the surveillance cameras.

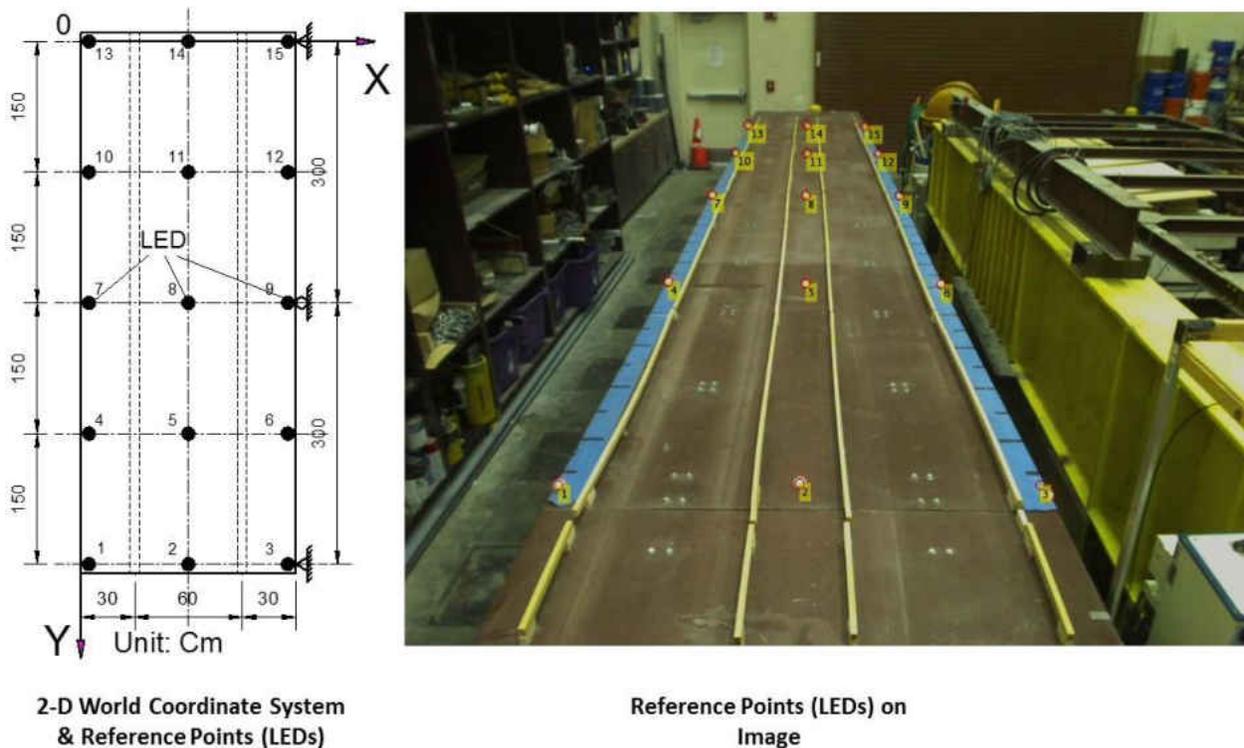


Figure 45. Reference points on the 2-D bridge deck coordinate system (left) and on the 2-D image coordinate system (right)

Since both the real life (X_i, Y_i) and image (u_i, v_i) coordinates are obtained of the reference points (LEDs) , these are substituted into Equation 43; and we have

$$\begin{bmatrix} X_1 & Y_1 & 1 & 0 & 0 & 0 & -X_1u_1 & -Y_1u_1 \\ 0 & 0 & 0 & X_1 & Y_1 & 1 & -X_1v_1 & -Y_1v_1 \\ & & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_i & Y_i & 1 & 0 & 0 & 0 & -X_iu_i & -Y_iv_i \\ 0 & 0 & 0 & X_i & Y_i & 1 & -X_iv_i & -Y_iv_i \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{bmatrix} = \begin{bmatrix} u_1 \\ v_1 \\ \vdots \\ u_i \\ v_i \end{bmatrix} \quad (45)$$

$$\text{Or} \quad X_{(2i,8)} a_{(8,1)} = u_{(8,1)}$$

$$\text{Where:} \quad X = \begin{bmatrix} X_1 & Y_1 & 1 & 0 & 0 & 0 & -X_1u_1 & -Y_1u_1 \\ 0 & 0 & 0 & X_1 & Y_1 & 1 & -X_1v_1 & -Y_1v_1 \\ & & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_i & Y_i & 1 & 0 & 0 & 0 & -X_iv_i & -Y_iv_i \\ 0 & 0 & 0 & X_i & Y_i & 1 & -X_iv_i & -Y_iv_i \end{bmatrix} \quad (46)$$

The Equation 46 is a form that can be solved by using the Linear Least Squares method. For this particular practice where 15 reference points are used, X becomes matrix of size $(30, 8)$, a and u columns of size $(8, 1)$. The matrix of a is calculated by following Equation.

$$a = (X^T X)^{-1} X^T u \quad (47)$$

$$\text{Or} \quad a = \begin{bmatrix} 14.13 \\ -1.00 \\ 0.00 \\ 8.99 \\ 109.30 \\ 0.14 \\ -13481.98 \\ -15872.12 \end{bmatrix} \quad (48)$$

$$\text{And} \quad T = \begin{bmatrix} 14.13 & -1.00 & 0.00 \\ 8.99 & 109.30 & 0.14 \\ -13481.98 & -15872.12 & 1 \end{bmatrix} \quad (49)$$

Once the matrix T is determined, a point on the bridge deck defined by the image coordinate system can be converted to its real-life estimated location defined by bridge deck coordinate system using the following relation:

$$\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = [T]^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (50)$$

The verified coordinates of LEDs obtained from the image coordinates and the matrix T are shown in Table 9. It is seen that coordinate errors alongside of the bridge width (120 cm) are from 0.00 cm to 1.46 cm, while coordinate errors alongside of the bridge length (600 cm) are from 0.00 cm to 7.51 cm. The results yield the maximum relative error of 1.2 %, which is acceptable for estimating locations of moving loads.

Table 9: Reference point coordinates: the world coordinate and the image coordinate systems (Column 2-5). The world coordinates calculated by the matrix T and errors (Column 6-9)

LED #	X (cm)	Y (cm)	u (pixel)	v (pixel)	X_{verified} (cm)	Y_{verified} (cm)	Error_X (cm)	Error_Y (cm)
C1	C2	C3	C4	C5	C6	C7	C8	C9
1	0	600	518.00	685.75	0.00	600.00	0.00	0.00
2	60	600	929.09	683.04	59.99	600.00	0.01	0.00
3	120	600	1344.91	688.45	121.46	602.02	-1.46	-2.02
4	0	450	722.50	369.81	0.96	456.07	-0.96	-6.07
5	60	450	939.89	370.86	60.24	456.46	-0.24	-6.46
6	120	450	1151.69	372.25	118.54	457.17	1.46	-7.17
7	0	300	796.52	247.93	0.01	300.14	-0.01	-0.14
8	60	300	943.33	248.63	60.19	299.69	-0.19	0.31
9	120	300	1087.83	249.78	119.96	300.15	0.04	-0.15

LED #	X (cm)	Y (cm)	u (pixel)	v (pixel)	X _{verified} (cm)	Y _{verified} (cm)	Error_X (cm)	Error_Y (cm)
C1	C2	C3	C4	C5	C6	C7	C8	C9
10	0	150	833.22	191.15	0.32	157.42	-0.32	-7.42
11	60	150	944.94	192.07	60.17	157.51	-0.17	-7.51
12	120	150	1055.51	191.38	120.27	152.27	-0.27	-2.27
13	0	0	856.53	153.06	-0.21	0.03	0.21	-0.03
14	60	0	946.79	153.06	60.63	-4.28	-0.63	4.28
15	120	0	1035.43	153.52	121.20	-6.14	-1.20	6.14

Conclusions

This chapter demonstrates a framework for obtaining loads induced by traffic vehicles passing over a bridge for BHM based only on computer vision techniques. The implementation consists of two main objectives: (1) to estimate vehicle loading amplitudes and (2) to localize vehicle positions. The highlight of the proposal framework is that neither bridge closure nor any conventional sensors are needed. Briefly, a surveillance camera is placed at a relatively higher position on the bridge for acquiring images of traffic vehicles moving on the deck. Those images are analyzed to identify vehicles and then categorize them into classes based on series of advanced computer vision algorithms and features including AdaBoost technique, cascade classifier and HOG descriptors. The detected vehicles are then labeled by bounding boxes corresponding to their types. Eventually, the positions of these bounding boxes (pixel metric) in image coordinates are converted to the positions defined by real-life coordinate system affixed on the bridge deck for localizing vehicle positions. The transformation process is achieved by adjusting the camera model equation based on the fact that vehicles move on a planar bridge deck.

The proposed approach is validated through tests conducted in the UCF Structural Lab on an experimental bridge. Multiple small-scale vehicles are deployed to simulate traffic on the bridge. Those vehicles are identified and categorized into their classes (*Class 1*, *Class 2*, or *Class 3*) at the detection rates ranging from 87.2 % to 95.9 % for all vehicle configurations. Subsequently, the detected vehicles are also localized by using the transformation matrix T , which is determined by means of employing 15 reference points on the bridge deck. The transformation matrix is verified for re-calculation of reference point locations in the world coordinate system from their places on images. The accuracy of reference point coordinates yield the maximum relative error of 1.2 %.

The process proposed in this Chapter is conducted and verified under laboratory conditions. Although the introduced framework shows fair results in laboratory, numerous considerations should be studied and addressed as to real-life implementations. Such considerations involve the task to develop a good database of vehicle classes as well as to identify real vehicles with very diverse appearances. The outcomes of the proposed methods are found acceptable and conveyed to the next tasks of this dissertation research in Chapter 4.

CHAPTER FOUR: OBTAINING UNIT INFLUENCE SURFACES FROM VISION BASED MEASUREMENTS: A NEW STRUCTURAL DAMAGE INDICATOR

Introduction

A very basic definition for Structural Identification (St-Id) is the development of a mathematical model to characterize input-output behaviors of a structure by analyzing experimental data. Common practice is to first identify, and then to classify these behaviors for assessment and decision making. Determination of damage or non-damage state, quantification of damage as well as locations of damage occurrences are some instances of this practice. Similar to any classifiers, a St-Id system makes use of various features, sometimes termed as *structural damage indicators* (or just *damage indicators* as in the context of this study) that are generated establish input-output interactions and later used to detect any change over time that can be related to damage. To successfully achieve this objective, a damage indicator has to provide evidence when there is variation from the established or defined healthy condition of a structure. For example, the curvature of a mode shape may be highly sensitive to discontinuity such as local cracking at a point and can be an effective damage indicator. Generally, damage indicators are determined by interpreting the measured data through signal processing and statistical pattern recognition techniques. Although numerous St-Id systems have been proposed for the last several decades, damage features can be categorized into two groups namely non-parametric based and parametric based indicators.

In non-parametric approach, damage indicators are extracted directly from SHM data in time domain. There is no information requirement associated with structural model such as geometric, material, and even loads as these methods only require the measured data from a sensor

network. In this group of methods, variety of advanced statistical analyses are utilized on time domain datasets to yield some sorts of statistical coefficients termed as non-parametric damage indicators, which must be roughly stable for a particular structural health condition. In case of acceleration data used as monitoring responses, Auto-Regressive (AR) based methods seem dominant (Nair, Kiremidjian, & Law, 2006; Omenzetter & Brownjohn, 2006; Sohn, Czarnecki, & Farrar, 2000; Sohn & Farrar, 2001). Originated from statistical model used in econometrics, an AR model describes the evolution of time series variables as a linear function of their past values. Since coefficients of the linear function show consistency and are calculated by least squares algorithm, a feature vector comprising those coefficients can be used as a damage indicator. An extended version of AR model is also developed since the current value of a time series can be predicted not only from its past values but also from other time series (exogenous inputs), for example the data from neighbor sensors. Examples of such models termed as Auto-Regressive with exogenous input (ARX) are found in numerous papers (Gul & Catbas, 2011; Lu & Gao, 2005; Q. Zhang, 2007).

Another group of non-parametric methods is developed by means of correlation based analysis. As a sensor network is commonly employed in a SHM study, those sensors are acquired at the same time and be also synchronized. It is expected to see that the measured data sets from various sensors in the network correlate with each other at different levels. Inspired from that observation, a feature vector of correlation coefficients among sensors are practiced as a sort of damage indicator. Some correlation methods are implemented on strain and deflection data such as Cross Correlation Analysis (CCA) (F. N. Catbas, Gokce, & Gul, 2012; Posenato, Lanata, Inaudi, & Smith, 2008), Robust Regression Analysis (RRA) (Laory, Trinh, & Smith, 2011; Posenato, Kripakaran, Inaudi, & Smith, 2010), Principal Component Analysis (PCA) (Posenato et al., 2008).

Although non-parametric damage indicators have promising advantages of simple implementation, no need for structural information and fast response for damage warning, those indicators are absolutely meaningless when it comes to physical interpretation of damage. Therefore, it is ineffective to utilize non-parametric indicators for damage evaluation as well as structural performance prediction for the future. Moreover, that type of indicators cannot be used to perform finite element model updating to aid users supplementally for understanding about non-instrumentation regions on structures.

On the other hand, parametric damage indicators are sorts of modal parameters derived from only structural characteristics such as mass, damping and stiffness. Hence, whenever condition of a structure changes, these indicators are altered indicating problems on the structure. Damage indicators in this group are exemplified as natural frequencies, mode shapes, curvature shapes, modal flexibility, and damping ratios that are obtained from dynamic testing or are attained later by analyzing static measured data such as strain and/or displacement unit influence line (UIL). In case of using dynamic data, these parametric indicators or so-called modal parameters are determined by handling Modal Analysis techniques, a powerful approach initially employed in aerospace from 1970s, and then mechanical and civil engineering. Since the approach has a long history, a large pool of research efforts has been implemented (Doebbling, Farrar, Prime, & Shevitz, 1996).

Conventionally, modal analysis based methods can be assorted into two main branches namely Experimental Modal Analysis (EMA) and Operational Modal Analysis (OMA). In EMA, damage indicators are estimated by means of input excitation-output dynamic response forms. Since it is quite challenging and expensive to excite large scale civil structures such as highway bridges, using EMA is not of interest in this dissertation research point of view. Conducting OMA

methods are either preferred or the only choice for data collection for identification due to the fact that this technique does not require input data. In OMA, the modal parameters are identified by processing vibration data of structures induced by ambient effects like wind and traffic. Peak Picking method is perhaps the most original OMA implementation to determine modal frequencies of structures after the transformation of ambient vibration data measured by sensors into frequency domain by variable techniques. This technique is conducted in this dissertation at Chapter 2 as well. Despite being simple and fast, the Peak Picking method is only suitable for a structure having well-separated modes and low damping ratios. Another method termed Frequency Domain Decomposition (FDD) is firstly introduced by Brincker et al. that based on Peak Picking technique and Singular Value Decomposition (SVD) (Brincker, Zhang, & Andersen, 2001). Random Decrement analysis is conducted in numerous researches (Gul & Catbas, 2008, 2009; Rodrigues, Brincker, & Andersen, 2004) for the reason that it is able to cleanly discard noise from the vibration data for further data processing steps. There are also other methods such as Stochastic Subspace Identification (SSI) (Ren, Zhao, & Harik, 2004; Van Overschee & De Moor, 2012), Complex Mode Indicator Function (CMIF) (F. Catbas et al., 1997; F Necati Catbas, Brown, & Aktan, 2004), to name a few.

Although obtaining parametric damage indicators by means of modal analysis has been employed in SHM since long time, this approach still gets much attention from researchers as these parameters may be used for many assessment purposes. The shortcomings of using modal analysis is that it requires structural information associated with a dense array of accelerometers to produce a better warning of damage. Furthermore, the modal parameters such as frequencies, mode shapes, damping ratios etc. are not effective for evaluating severe level of damage or estimating structural remaining life-time, which are directly related to structural analyses.

To determine local damage with another approach for structural identification for input-output characterization, influence lines can be employed at critical elements as presented in superload case study on three highway bridges by Turer et al (Turer, Levi, & Aktan, 1998). Recently, obtaining unit influence line (UIL) from real-life measured data as a new type of damage indicator has earned considerable attention in Bridge Health Monitoring, an essential practice in SHM. Unlike modal parameters, strain and/or deflection unit influence lines are explicit structural features that are directly related to responses adopted to structural valuation and assessment. Hence, these parameters are quite intuitive for civil engineers. Obtaining an Unit Influence Line (UIL) from measurements is introduced by Catbas and Aktan along with other promising indices for condition and damage assessment with real life examples of utilization (F Necati Catbas & Aktan, 2002). A study for obtaining an influence line from measurement is introduced by OBrien et al. (O'Brien, Quilligan, & Karoumi, 2006). Following this algorithm, strain influence lines could be calculated by knowing vehicle axle weights as well as axle internal distances. As measured responses are superimposed from all axles, a mathematical method is proposed to decompose the influence of each axles. Since these influences are derived from ordinates of the bridge UIL, the vector of those ordinates is obtained by solving multiple equations corresponding to each scans of the data. The UILs determined by this method are conducted as damage indicators in successor papers (González, Rowley, & OBrien, 2008; OBrien, Carey, & Keenahan, 2015). Another implementation for Bridge Health Monitoring using UIL is conducted by Stohr et al. (Stöhr, Link, Rohrman, & Rümer, 2006). In this research, incline (tilt) UILs are attained at a support by placing quaci-static load along a steel beam on a laboratory. The test results show that these incline UILs are pretty sensitive to some simulated damage on the beam. Orcesi and Frangopol conduct a study employing strain UILs for analyzing life-time structural reliability of a long span bridge (Orcesi

& Frangopol, 2010). The strain UILs herein are determined from crawl tests, which are commonly conducted by slowly driving a pre-weighted vehicle (so-called a testing vehicle) on a bridge deck following a pre-assigned path. Correlation analysis such as moving principle component analysis (MPCA) and robust regression analysis (RRA) on measured displacement and tilt UILs are studied by Cavadas et al. (Cavadas, Smith, & Figueiras, 2013). The correlation parameters gained from that implementation illustrates capability of damage detection on a frame structure. Using UIL as a damage indicator is also preferred on long cable bridges such as cable-stayed and suspension bridges (Z.-W. Chen, Zhu, Xu, Li, & Cai, 2014; Zhu, Xu, & Xiao, 2014). Since cable bridges are very complex structures consisting of cables, trusses, beams etc., it is very difficult to extract modal parameters for assessment purposes. By capturing strain UILs as a result of deploying crawl tests or using train (metro shuttle) loading, some induced damage could be identified in these monitoring studies. An alternative approach for determining UIL without using a pre-weighted vehicle is proposed by Catbas and Zaurin (F. Necati Catbas et al., 2012; R. Zaurin & Catbas, 2010). In these studies, the authors try to utilize a traffic vehicle instead of a testing truck. The information of a traffic truck such as type of vehicle, weight, and locations on a bridge deck is estimated by means of computer vision techniques. The UILs determined in these papers are employed not only for damage detection but also for bridge load rating.

Motivation and Objectives

Obtaining a sensitive damage indicator for engineering decision making is an ultimate aim for Structural Health Monitoring studies since this feature can be employed towards safety, serviceability, maintenance and operational decision in relation to the “*the health condition*” of a particular structure. After three decades of practice in SHM, a large pool of damage indicators has

been introduced, obtained and then utilized to identify damage in structures, especially for bridges. Despite only the highlight research belonging to the field are reviewed in this Chapter, accomplishments about damage indicators are compiled as subsequent bullets

- ✓ Non-parametric damage indicators have been receiving attention from SHM community lately due to model-free analysis enabling fast and simple implementation. The methods for extracting non-parametric indicators are substantially based on statistical algorithms such as Auto-Regressive (AR) and Correlation based analyses. Since the damage indicator is a type of statistical coefficient, it is independent from any structural parameter. Thus, non-parametric indicators do not help users measure damage level as well as update finite element model for further understanding of structural behavior.
- ✓ On the contrary, modal parameters, sorts of parametric damage indicators, have been maturely studied for the last several decades. These parameters are obtained from two main approaches including Experimental Modal Analysis (EMA) and Operational Modal Analysis (OMA). OMA method is preferred in civil engineering due to no loading information demanded. Modal analysis requires structural information and also a dense network of accelerometers to yield better signs of damage. In civil engineering, deploying modal analysis is not easy for quantifying severe level of damage and also estimating remaining life-time of structures, which is commonly governed by structural analysis.
- ✓ Recently, Unit Influence Lines (UILs) are practiced as new type of damage indicators especially in Bridge Health Monitoring (BHM). Since UILs are structural parameters, they are capable of addressing some shortcomings of both modal analysis parameters

and statistical analysis coefficients when acting like damage indicators. The weakness of developing UIL is that vehicle of a known-weight is required. In addition, since developing an UIL depends on the tests along truck pathways, the obtained UIL is only invariable if either vehicle testing routes are kept the same or the bridge distribution factors are well-established. Those shortcomings limit the practicing for UIL extraction in real-life.

In this Chapter, a new damage indicator is introduced as a consequence of coupled outcomes from previous Chapter 2 and Chapter 3. The proposed damage indicator is Unit Influence Surface (UIS), which is a 3-D construction of UIL. In bridge structural analysis, deploying UIS is more convenient than using UIL for calculating internal forces (e.g. moment, shear, and normal force) as well as responses (e.g. strain, deflection) since UIS based analyses do not require distribution factors. Similar to UIL, UIS is a normalized structural parameter influenced by only structural characteristics such as mass, stiffness, and geometric info. In other words, UIS can be employed as a possible damage indicator for bridge health monitoring. In this study, estimated traffic loading information including vehicle weights and positions (from Chapter 3) is coupled with measured displacements (from Chapter 2) to construct displacement UIS. Other types of UIS can be also obtained in the same manner corresponding to different types of measured responses such as strain and tilt (rotation). The details of theoretical background are described in next sections. Subsequently, the framework is validated on the small-scale bridge located in the UCF Structural Lab.

Theoretical Background

Theory of Unit Influence Surface (UIS)

Unit Influence Surface (UIS) of a certain response (displacement, strain, etc.) at a measurement location on a beam-type or plate-type structure (e.g. single or multi-span bridge with its deck) is defined as a response function of *the unit load* with respect to the any location on that structure. Since the value of *the unit load* equals to one (1), a UIS can be mathematically presented as a two-variable function S as shown in Equation 51.

$$U = S(X, Y) \quad (51)$$

where (X, Y) represents the location of *the unit load* on the bridge deck coordinate system. U which is called as the normalized response is the response at the measurement point due to *the unit load* at (X, Y) . An example of UIS is depicted in Figure 46.

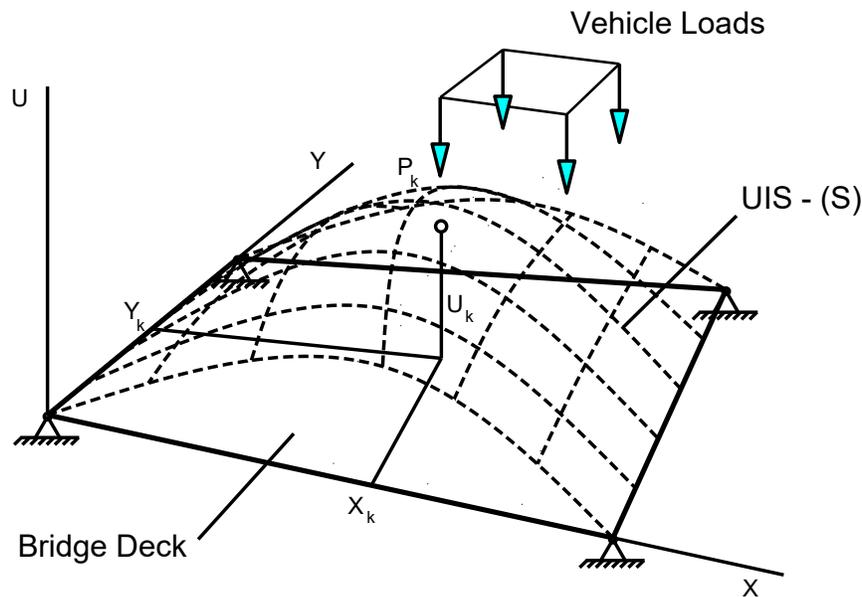


Figure 46. An example of Unit Influence Surface

As mentioned previously, a UIS is used to calculate internal forces and responses at a position on a bridge girder induced by moving loads such as vehicles passing over the deck. Let a

group of loads on a bridge be denoted by $\{P_1, P_2, \dots, P_k\}$, and corresponding locations by $\{(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)\}$ as illustrated in Figure 46. Then, the vertical coordinates of the UIS under those loads derived from Equation 51 are

$$U_k = S(x_k, y_k) \quad (52)$$

Thus, the internal force or response R (depending on type of UIS) can be determined as follows

$$R = \sum_1^k U_k P_k \quad (53)$$

Using UIS for structural analysis is quite convenient and simple, especially when dealing with movable loads. Moreover, a UIS is not function of the loading but the structural characteristics including geometry and stiffness, which makes UIS as a potential damage/change indicator. However, determining a UIS in real life following its definition is not a direct measurement for the reason that *the unit load* is an unreal effect. Hence, structural responses to *the unit load* cannot be measured. As a remedy, a practical approach is introduced for constructing UIS from direct measurements.

Construction of Unit Influence Surface (UIS) from Direct Measurement

In this study, displacement Unit Influence Surface (UIS) is developed from measured data. Obviously, other types of UIS derived from strain, tilt (rotation), and also other capably measured responses can be also obtained in the same manner proposed in this Chapter. Herein, the displacement UIS is constructed from direct measurements including vehicle axle weights, vehicle locations, and structural displacements (Figure 32). All of these measurement protocols are introduced in Chapter 2 (i.e. structural displacement measurement) & 3 (remaining vehicle-info

estimations). Briefly, the methodology for constructing UIS is proposed based on surface fitting algorithms for a set of 3-D discrete points called *UIS-points*. A UIS-point describes the state of *vehicle loading-structural response* (input-output) interaction. Intuitively, a certain state corresponds to an instant when a scene of traffic vehicles and corresponding bridge girder deflection due to those vehicles is captured by the surveillance camera. While two horizontal coordinates of a UIS-point are bridge-deck coordinates of *the configuration location* that are derived from vehicle positions, vertical coordinate of that UIS-point is *the normalized displacement* of the bridge girder. Details of determining *the configuration location* and also *the normalized displacement* from direct measurements are described in next sections.

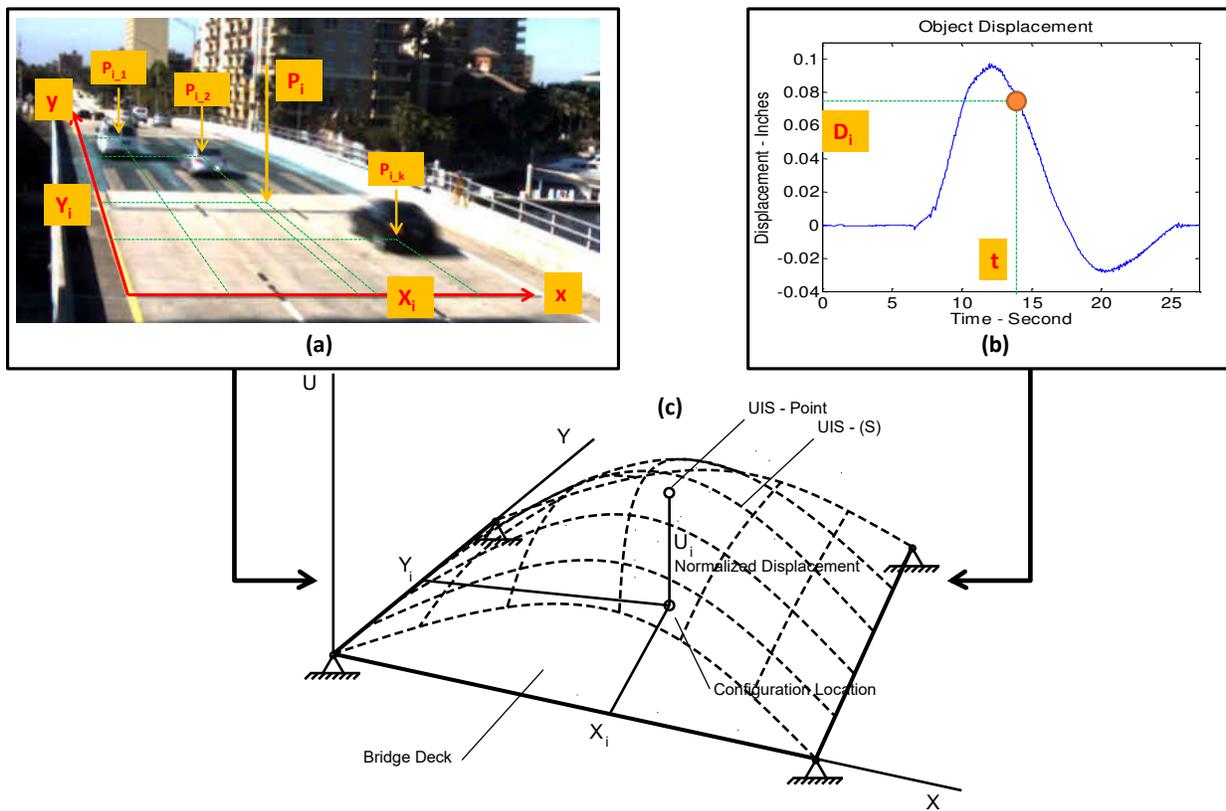


Figure 47. Protocol for extracting displacement Unit Influence Surface from direct measurements

The Configuration Location

The configuration location is the representation of equivalent location that accounts for a certain vehicle configuration (combination of multiple vehicle locations) on the bridge deck. At a particular instant, the scene of traffic vehicles is acquired by the surveillance camera. Then, those vehicles are identified by their weight and localized by their positions on the bridge deck (Chapter 3). For example, the estimated loads of each wheel are $\{W_1, W_2, \dots, W_j\}$ while the positions of these loads are $\{(x_1, y_1), (x_2, y_2), \dots, (x_j, y_j)\}$. In this study, the coordinates of the configuration location (X, Y) are subjectively appointed as the centroid of the load group, which can be determined as

$$X = \frac{W_1x_1 + W_2x_2 + \dots + W_jx_j}{W_1 + W_2 + \dots + W_j} \quad (54)$$

$$Y = \frac{W_1y_1 + W_2y_2 + \dots + W_jy_j}{W_1 + W_2 + \dots + W_j} \quad (55)$$

The Normalized Displacement

The normalized displacement U is defined as the vertical coordinates of the UIS at the configuration location (X, Y) . Assuming that the surface S is the two-variable function of the UIS that needs to be found, the measured displacement D induced by the vehicle configuration must be constrained with the function S following Equation 53.

$$D = W_1S(x_1, y_1) + W_2S(x_2, y_2) + \dots + W_jS(x_j, y_j) \quad (56)$$

On the other hand, another constrain is expressed as follows:

$$D = W^{equiv}U \quad (57)$$

$$\text{Similarly: } D = W^{equiv}S(X, Y)$$

Where W^{equiv} is an equivalent load placed at the configuration location.

Equations 56 and 57 yields

$$W^{equiv}S(X, Y) = W_1S(x_1, y_1) + W_2S(x_2, y_2) + \dots + W_jS(x_j, y_j) \quad (58)$$

Or

$$W^{equiv} = W_1 \frac{S(x_1, y_1)}{S(X, Y)} + W_2 \frac{S(x_2, y_2)}{S(X, Y)} + \dots + W_j \frac{S(x_j, y_j)}{S(X, Y)} \quad (59)$$

Set $r_j = \frac{S(x_j, y_j)}{S(X, Y)}$ is the weighted-factor of the load W_j in the way of constructing the equivalent load W^{equiv} . Equation 59 becomes

$$W^{equiv} = W_1r_1 + W_2r_2 + \dots + W_jr_j \quad (60)$$

Thus, it is seen that the normalized displacement U can easily be calculated if the equivalent load W^{equiv} is obtained (Equation 57). Unfortunately, determining the equivalent load W^{equiv} by means of Equation 60 is not a straightforward process because all the weighted-factors r_j are unsettled. The solution to tackle this challenge is an iterative approximation algorithm, as shown in the following, which makes use of many relationships in the same form of Equation 60 derived from many vehicle configurations, respectively.

- ✓ Step 1: Given n states of vehicle loading-structural response (input-output) interaction, there will be n values of measured structural displacements D_i . There are also n vehicle configurations providing n configuration locations $\{(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)\}$. For a particular state, vehicle loads $\{W_1, W_2, \dots, W_j\}$ and vehicle positions $\{(x_1, y_1), (x_2, y_2), \dots, (x_j, y_j)\}$ are obtained. Please note that the number of loads (or number of detected vehicles) j in each vehicle configuration varies.
- ✓ Step 2: Initialize all weighted-factors $r = 1$.
- ✓ Step 3: For $i = 1, \dots, n$. Operating for each states
 - Calculate W_i^{equiv} for each vehicle configurations following Equation 60.

$$W_i^{equiv} = \sum_1^{j_i} W_{j_i} r_{j_i}$$

- Obtain the normalized displacement U_i (Equation 57)

$$U_i = \frac{D_i}{W_i^{equiv}}$$

- ✓ Step 4: Utilize surface fitting algorithm for U_i , which yields the surface function S .
- ✓ Step 5: Update all weighted-factors based on the function S

$$r_j = \frac{S(x_j, y_j)}{S(X, Y)}$$

- ✓ Loop through the Step 3, Step 4, and Step 5 until the error is less than a pre-established threshold. The error value is calculated as the percentage difference of the S volumes between two consecutive fitting implementations. In this study, 1% of difference is selected as the threshold to stop the iteration.

Surface Fitting for Discrete Data

Surface fitting is the core analysis for constructing UIS in this study. In general, surface fitting is the process to develop a 2-D surface that fits the best to a set of 3-D points. Since the 2-D surface is a mathematical function with two variables and numerous of constant parameters, that fitting surface is regulated if its function type and also corresponding parameters are determined. The common procedure to construct a fitting surface is to hypothesize a mathematical function, and then to estimate its parameters to achieve the best fit with a given discrete data set. Due to the hypothesized mathematical function being subjectively appointed by users, the approach for its selection is an iterative process based on evaluating the accuracy of fit such as SSE (sum of squares due to error), R-square (coefficient of determination), and RMSE (root mean squared error).

Although numerous mathematical functions and methods for obtaining function parameters are introduced, these can be assorted into two groups. In the first group, the mathematical function is formed from a single equation, for example, a polynomial, a trigonometric or a distributional equation. The second group involves either interpolation or smoothing function, which are combinations of finite equations. While the first group of fitting functions can provide parameters for the whole set of data, they require well-understood behavior of experimental data sets and also are highly influenced by outliers. On the other hand, in spite of no parameter extraction, interpolation and smoothing functions yield better matching to given discrete data sets and do not require deep knowledge of the data shapes. Knowing that a UIS should be a complex function involving many characteristics (mass, stiffness, geometric, etc.) of a composite structure, it is impossible to identify a mathematical UIS function. Thus, the approach of using interpolation and smoothing functions for fitting (so-call nonparametric fitting) is deployed in this dissertation study. The selected methods are 1) Bilinear Interpolation and 2) Thin-plate Spline Interpolation fitting that are briefly reviewed later on.

Bilinear Interpolation

Bilinear interpolation is the simplest and the most straightforward method in the pool of interpolation. As an extension of linear interpolation for two variables, it can be deployed by performing linear interpolation in each variable, consecutively. In mathematics, the surface function S developed by means of bilinear interpolation comprises finite equations, each of which is represented as in the following form

$$f(X, Y) = a_0 + a_1X + a_2Y + a_3XY \quad (61)$$

A surface segment $f(X, Y)$ is determined from four adjacent data points; and it is only valid inside the space limited by these points. For example, given four adjacent UIS-points such as (X_1, Y_1, U_1) , (X_2, Y_2, U_2) , (X_3, Y_3, U_3) , and (X_4, Y_4, U_4) , the coefficients (a_0, a_1, a_2, a_3) are calculated by substituting the given values into Equation 61, and then solving the equation below.

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1 & X_1 & Y_1 & X_1Y_1 \\ 1 & X_2 & Y_2 & X_2Y_2 \\ 1 & X_3 & Y_3 & X_3Y_3 \\ 1 & X_4 & Y_4 & X_4Y_4 \end{bmatrix}^{-1} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \quad (62)$$

Since every surface segment is obtained from four certain adjacent 3-D points in the data set, the fitting surface is a combination of these segments. An example result for constructing a surface from a discrete data set (X, Y, U) in terms of the bilinear interpolation is illustrated in Figure 48. It is seen that the fitting surface passes through every given data points (blue points). That makes the surface rough with hill-peaks. Using bilinear interpolation returns a surface reflecting original information, but it also catches all outliers. One advantage of deploying bilinear interpolation is that the method performs efficient computation because of its simple methodology. In this study, bilinear interpolation is employed in *Step 4* in the iteration algorithm for obtaining the UIS described in the previous section.

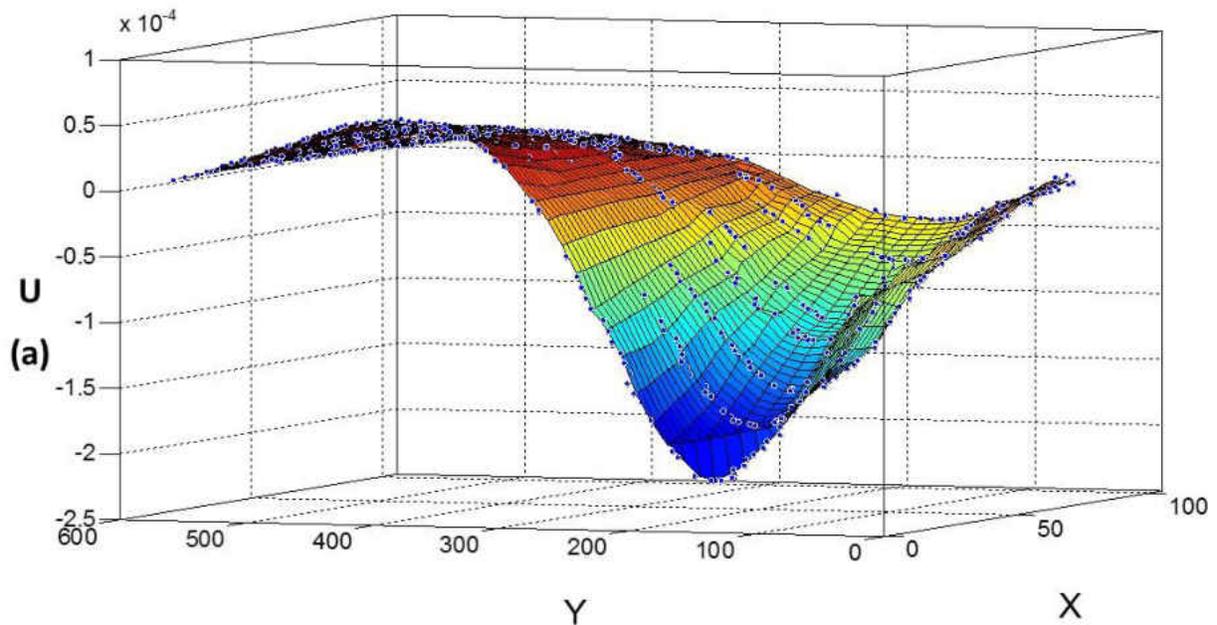


Figure 48: A fitting surface constructed by using the bilinear interpolation

Thin-plate Spline Interpolation

In general, a spline is a combination of multiple mathematical curves that are connected through predefined points (so-called *knots*). Due to the smooth shape of a spline by its construction, the two-first derivatives of the spline must be continuous everywhere, apparently, at the *knots* as well. Hence, a spline interpolation is commonly carried out in terms of these continuous derivative constrains and also a type of math function selected for its segments. Since a spline is merged by many segments, the spline based interpolation error can be achieved even using a simple math form for segment itself. For example, Figure 49 shows that a spline (red color) derived from a 3rd order polynomial function is well-fitted to a given dataset (blue dots) and better than a 7th order polynomial function (black color).

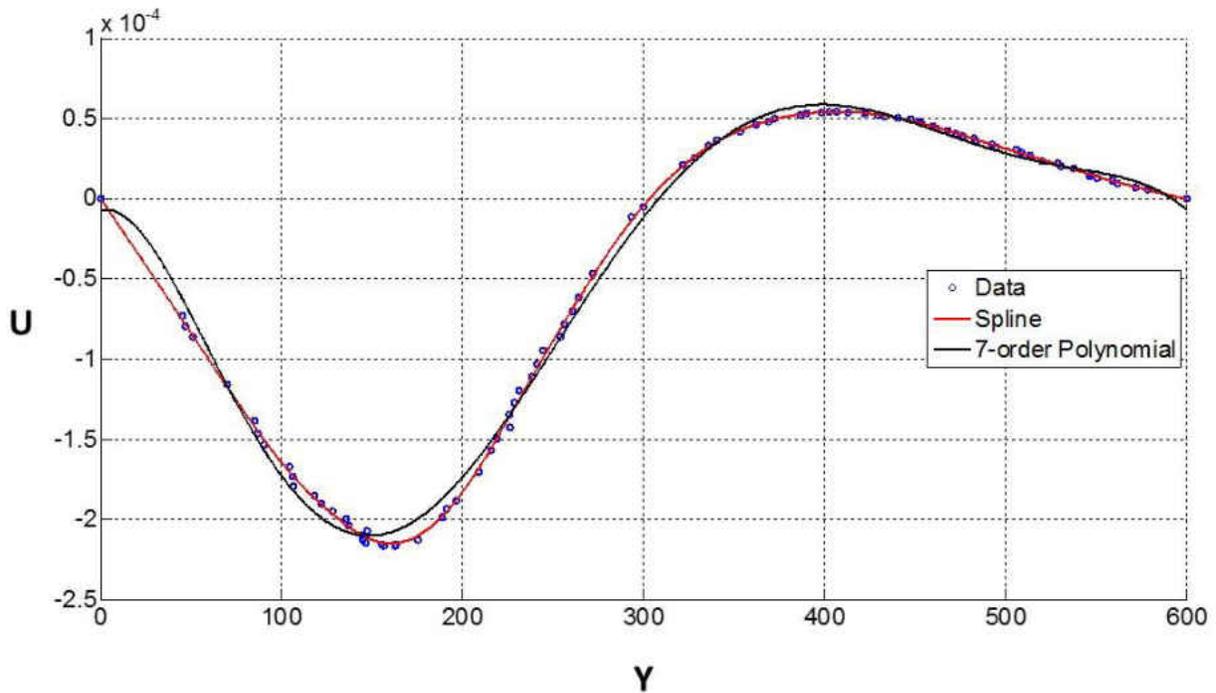


Figure 49: Spline interpolation vs. high order polynomial interpolation

Multi-variable spline interpolation is an extended version done by conducting the spline interpolation on multiple dimensions. In this group, thin-plate spline interpolation is one of the most popular data fit technique for two dimensional datasets, which is initially introduced by Duchon (Duchon, 1977). Inspired by a physical observation involving the bending of a thin sheet of metal that has a certain rigidity, the smoothing criteria of the thin-plate spline can be controlled based on a rigid parameter. By assuming that the rigidity is equal along the whole thin-plate, the thin-plate spline interpolation yields a better plausible fitting surface, especially for physical engineering problems. This assumption also helps to discard outliers effectively making the thin-plate spline interpolation preferable in many applications such as data mining, 3-D construction of geometric models, image warping etc., to name a few.

The fundamental function used to model a thin-plate spline is a sort of *radial basis function* as shown in Equation 63.

$$\Psi(r) = r^2 \ln(r) \quad (63)$$

where r is the Euclidean radial length (distance from a data point to the origin) of a vector comprising independent variables. For the case of two independent variables (X, Y), the Equation 63 can be rewritten as Equation 64. It is also seen that the function Ψ has its valid derivatives at the two-first orders.

$$\Psi(X, Y) = (X^2 + Y^2) \ln(\sqrt{(X^2 + Y^2)}) \quad (64)$$

Then, a simple form of the thin-plate spline interpolation can be described as a combination of the radial basic Ψ functions and a first-order polynomial part.

$$f(X, Y) = \sum_{i=1}^{n-3} \Psi((X, Y) - c_i) a_i + X a_{n-2} + Y a_{n-1} + a_n \quad (65)$$

where c_i is the control point in \mathbf{R}^2 (*knots*); and a_i is the parameter that is calculated by minimizing the energy function E :

$$E = \sum_{j=1}^{n-3} \|U_j - f(X_j, Y_j)\| + \gamma \iint \left[\left(\frac{d^2 f}{d^2 X} \right)^2 + 2 \left(\frac{d^2 f}{dX dY} \right)^2 + \left(\frac{d^2 f}{d^2 Y} \right)^2 \right] dX dY \quad (66)$$

Where (X_j, Y_j, U_j) are the data points (the UIS-points herein) for constructing the thin-plate spline; γ is the rigid parameter that is mentioned previously. It is also proved that minimization of Equation 66 returns a unique parameter set of a_i ; or a single f function, in other words (Wahba, 1990).

A fitting surface obtained by means of deploying the thin-plate spline interpolation is depicted in Figure 50b. Also, a comparison between fitting surfaces of the same discrete dataset constructed by both methods (bilinear interpolation and thin-plate spline interpolation), is also

illustrated in Figure 50. The figure shows that all hill-peaks on the bilinear interpolated surface are smoothed on the thin-plate spline surface. Obviously, using the thin-plate spline interpolation yields a plausible outcome, especially when the data is measured from a rigid plate such as a bridge deck. However, it is seen that the bilinear interpolation is also capable of providing a good result if the measured data is processed discarding outliers. Since thin-plate spline method suffers from computational efficiency, this deployment is only utilized for constructing the final UIS for further analyses to identify damage that is presented in Chapter 5.

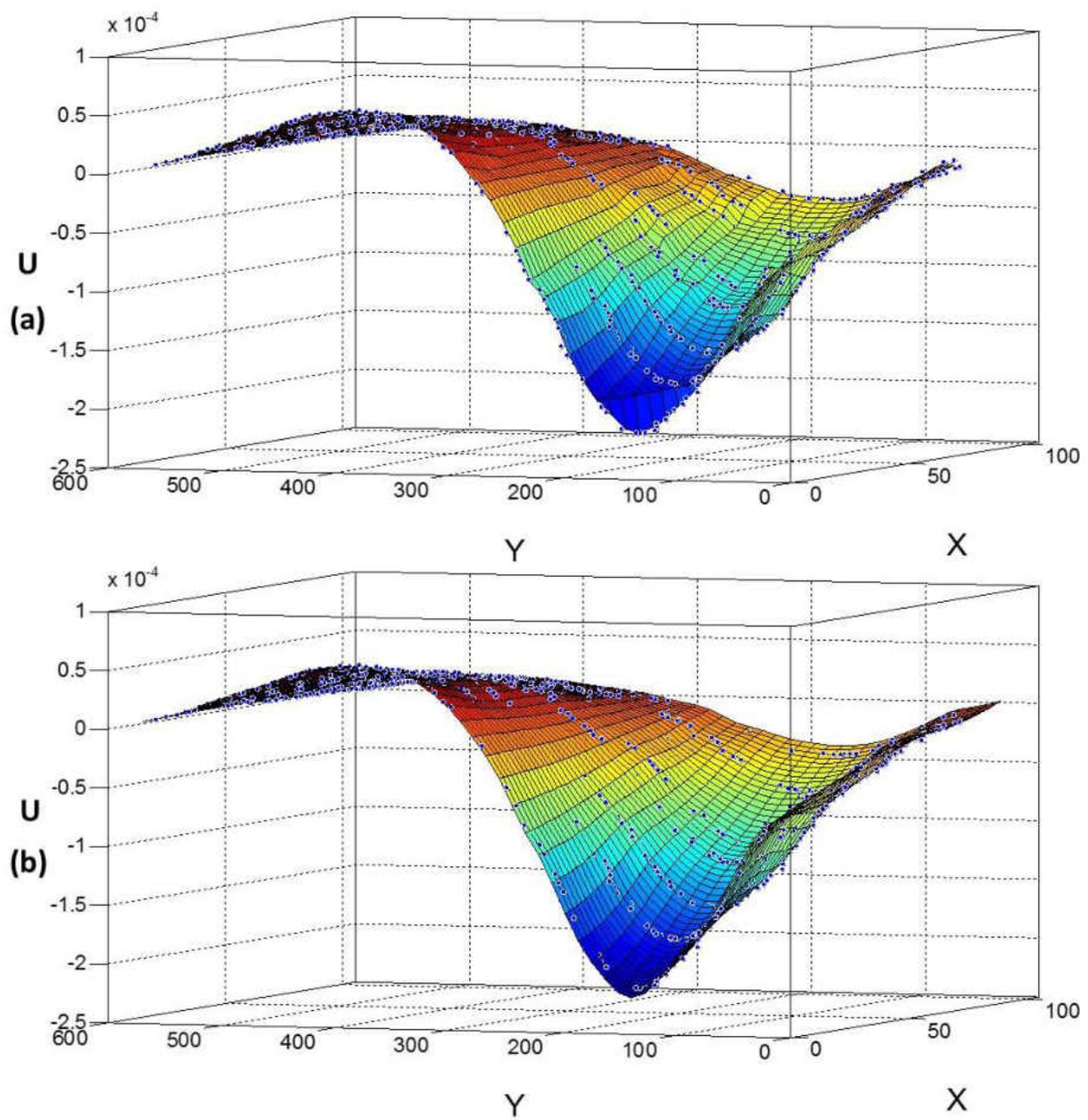


Figure 50: Fitting surfaces of a given dataset obtained by two selected methods: a) Bilinear interpolation; b) Thin-plate spline interpolation

Laboratory Verification

Experiment Setup on UCF 4-Span Bridge

The displacement Unit Influence Surface (UIS) is obtained for the UCF 4-Span Bridge: an SHM apparatus on the UCF Structural Lab detailed in the previous Chapters. To obtain the displacement UIS from direct measurements, two groups of data are needed to be acquired including traffic vehicle information and bridge girder displacements. Since the experimental setup for simulating traffic and for determining info of vehicles is described in Chapter 3, this section only focuses on the illustration of the experimental setup for measuring displacements of bridge girders caused by those simulated traffic configurations.

The displacement UIS's are extracted from a number of positions on UCF 4-Span Bridge girders in a comparative fashion and also localization of damage locations, which are introduced in the next Chapter. Since the bridge has two continuous spans consisting of two main girders, the measurement locations are assigned below the girder flanges and along these girders as depicted in Figure 51. The displacement of a measured location can be obtained by using the non-target vision based method as proposed in Chapter 2. To be able demonstrate UIS for several locations, the displacements from other twelve (12) monitored positions are obtained by a set of LVDTs. The vision based displacement method is also conducted at a location for verification as presented in Chapter 2. A National Instrument data acquisition system (NI-SCXI) is employed to acquire data from these LVDTs. To synchronize the image data of vehicles and the displacement data of the bridge girders, both the NI data acquisition and the webcam for capturing small-scale cars passing over the bridge deck are triggered from a shared LABVIEW code.

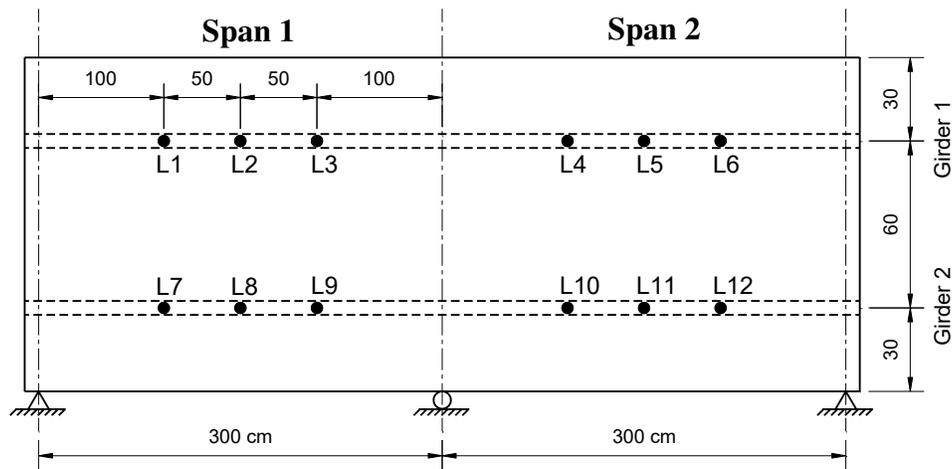


Figure 51: The twelve (12) measured locations on the UCF 4-Span Bridge girders

Unit Influence Surfaces Results and Discussion

As mentioned in the Chapter 3, a total of forty (40) crawling tests are conducted to get enough data for obtaining numerous UIS's for every measurement locations. For each test, the vehicles are driven one-turn (back and forth) from the starting point to the end of the bridge. Meanwhile, dynamic displacements caused by the vehicles are collected at the twelve (12) measurement locations simultaneously. As only static responses of dynamic displacements are of interest for constructing UIS, a low-pass filter (filtering off high frequencies from the raw data) is employed to discard the dynamic part of the raw data while keeping the static response. The static displacements that are extracted from dynamic raw data of the *L1* location at a particular test are depicted in Figure 52a. The filtering process is applied to data for all measurement locations. The static displacement results of several locations, for instance from *L1* to *L6* that are acquired simultaneously, are also shown in Figure 52b.

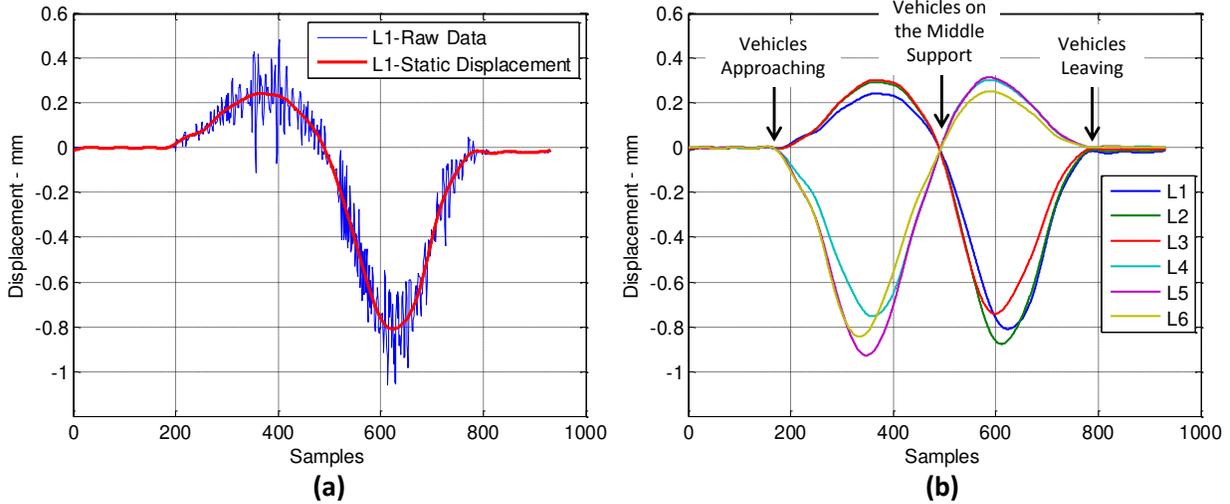


Figure 52: Experimental displacement data; a) The raw data and extracted static part at the location $L1$; b) The static displacements at the locations from $L1$ to $L6$

Since a UIS is constructed by surface fitting to a number of discrete data points, a combined data set from several tests is a need for the fitting analysis. To verify the consistency of the proposed UIS constructing algorithm, a number of fifteen (15) UIS's are developed for each measurement position by randomly selecting ten (10) tests out of forty (40) from the test database. A particular combination of data sets from 10 selected tests for constructing UIS's is named a *Set*, one of which is shown in Figure 53. As seen in this figure, each colored dot represents a *UIS-point* in the *Set*. The locations of these UIS-points in the plan view (X -bridge width, Y -bridge length) present vehicle configuration locations (ref. The Configuration Location section) while the *U-unit displacement* values of that color dot are the normalized displacements determined by means of the iterative algorithm (ref. The Normalized Displacement section). The plan view (Figure 53b) also illustrates that vehicle routes can be anywhere on the bridge deck; and these vehicles even do not travel straightly. Hence, it should be a convenience of using UIS instead of UIL for real-life bridges since constructing a UIL has a limitation of a certain pre-routed vehicle pathway.

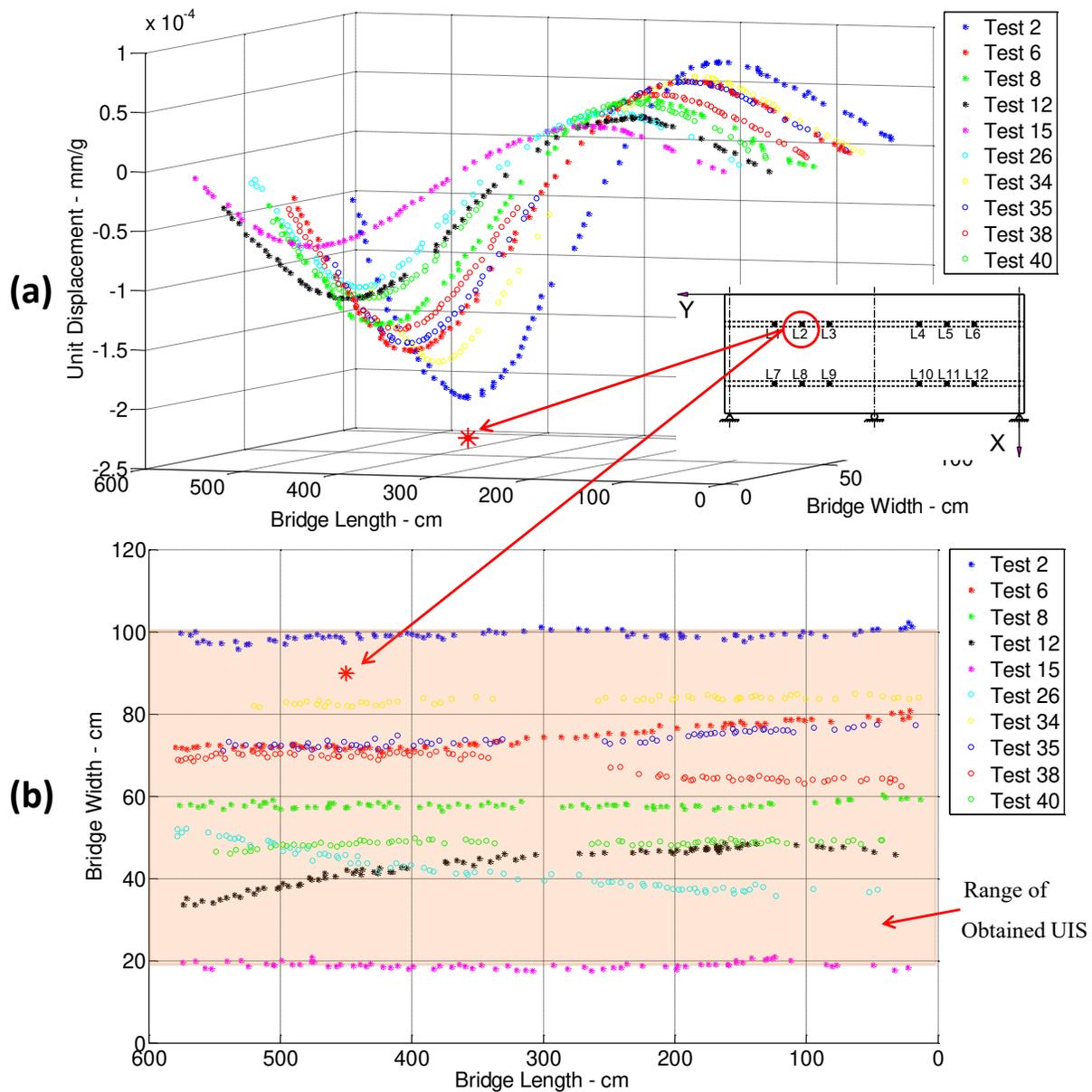


Figure 53: A combination data from the 10 tests acquired at the location L2 for constructing displacement UIS at the L2 (dUIS-L2); a) 3-D View; b) Plan View

The *UIS-points* (color dots) are employed to fit a surface by means of the thin-plate spline interpolation. Since the obtained *UIS-points* are commonly between 20 cm and 100 cm on the bridge width axis, only the surface range of 20-100 cm is constructed yielding a constructed UIS

(Figure 53b). Thus, the size of the UIS in the plan view becomes 80 cm x 600 cm in this particular study. The displacement UIS's at the locations $L7$, $L8$, and $L9$, denoted as $dUIS-L7$, $dUIS-L8$, and $dUIS-L9$ respectively, are depicted in Figure 55. Due to the twelve (12) measurement locations being symmetrically distributed, these symmetrical UIS's are expected to be similar in groups for example group **I** of $\{dUIS-L1, dUIS-L6, dUIS-L7, dUIS-L12\}$, group **II** of $\{dUIS-L2, dUIS-L5, dUIS-L8, dUIS-L11\}$, and group **III** of $\{dUIS-L3, dUIS-L4, dUIS-L9, dUIS-L10\}$ as shown in Figure 54. The obtained UIS's at locations $L1$, $L6$, $L7$ and $L12$ (group **I**) are illustrated in Figure 56.

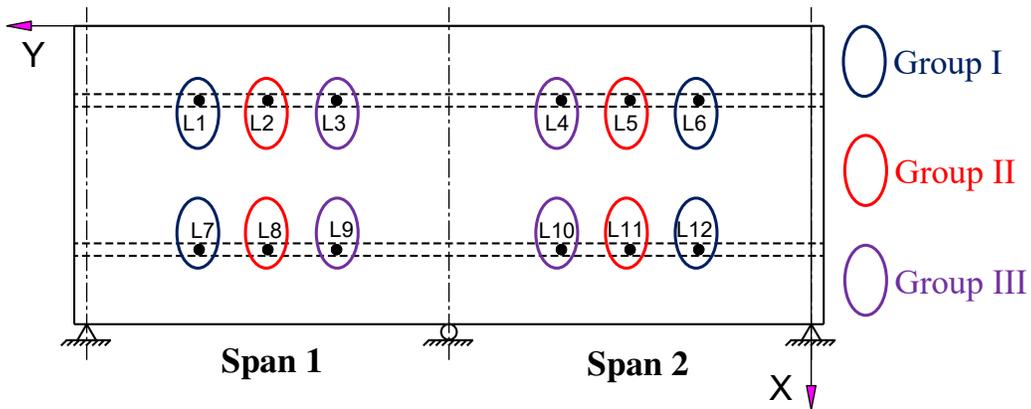


Figure 54: The measured locations have geometrical symmetry. The UIS's at the locations in the same group should be similar

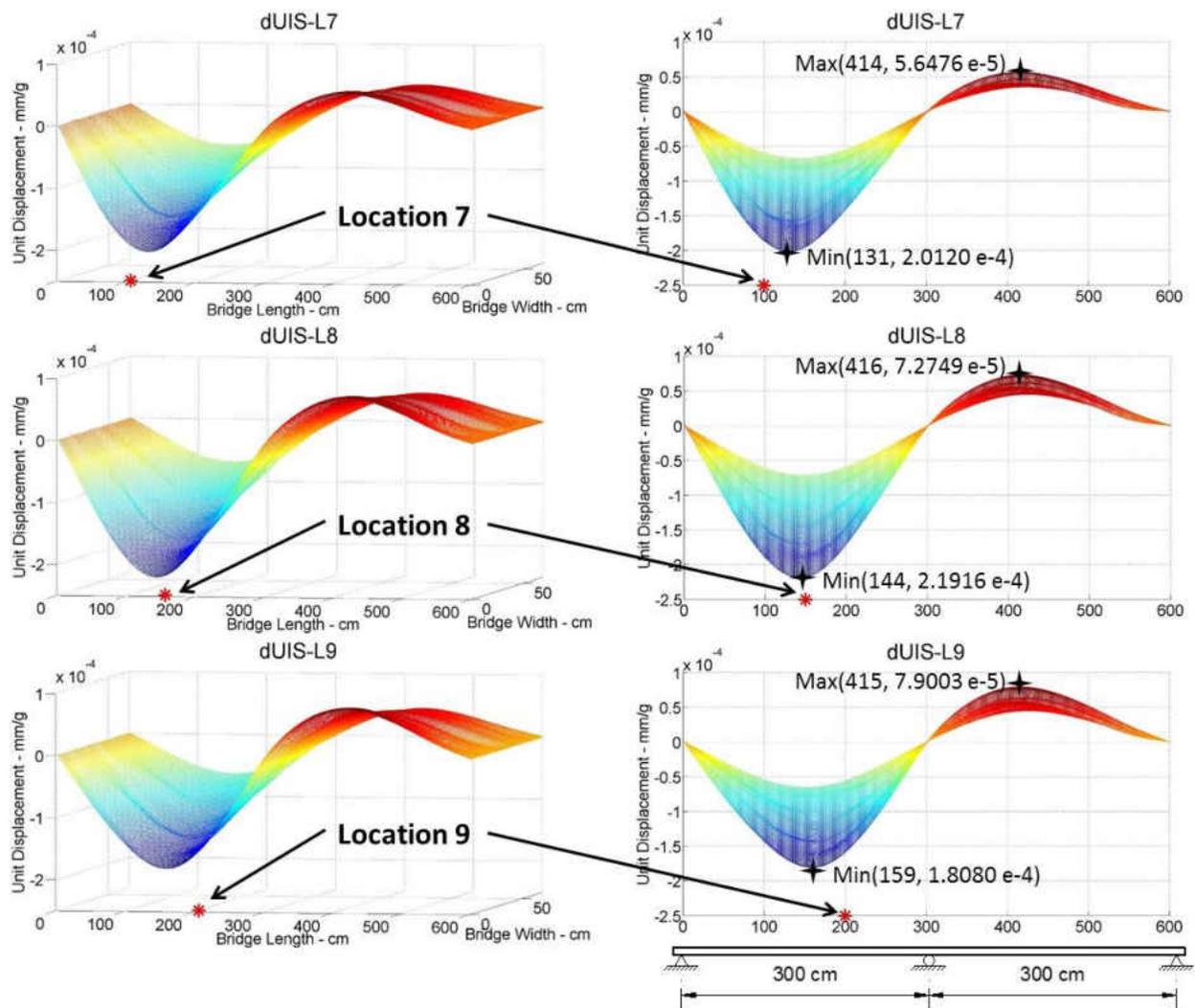


Figure 55: The displacement UIS's at the locations $L7$, $L8$, and $L9$

Figure 55 shows 3-D views (left column) and also side views (right column) of the UIS's from the locations $L7$, $L8$, and $L9$. As expected, all the UIS's go through zero-value points at the positions of the bridge supports (0, 300, and 600 cm on the bridge length axis). The maxima points on the *Span 1* of the UIS's are located at 131, 144, and 159 cm distances from the left-end of the bridge for the locations $L7$, $L8$, and $L9$ respectively. Also, the highest value of these UIS's is on the $dUIS-L8$. It is because the location $L8$ is the middle point of the span, which is commonly deflected the most in a beam. Related to other maxima points on the *Span 2*, the highest value

occurs on the *dUIS-L9*; and that is very plausible following structural analysis. However, the positions of these maxima points do not alter much for every measured locations.

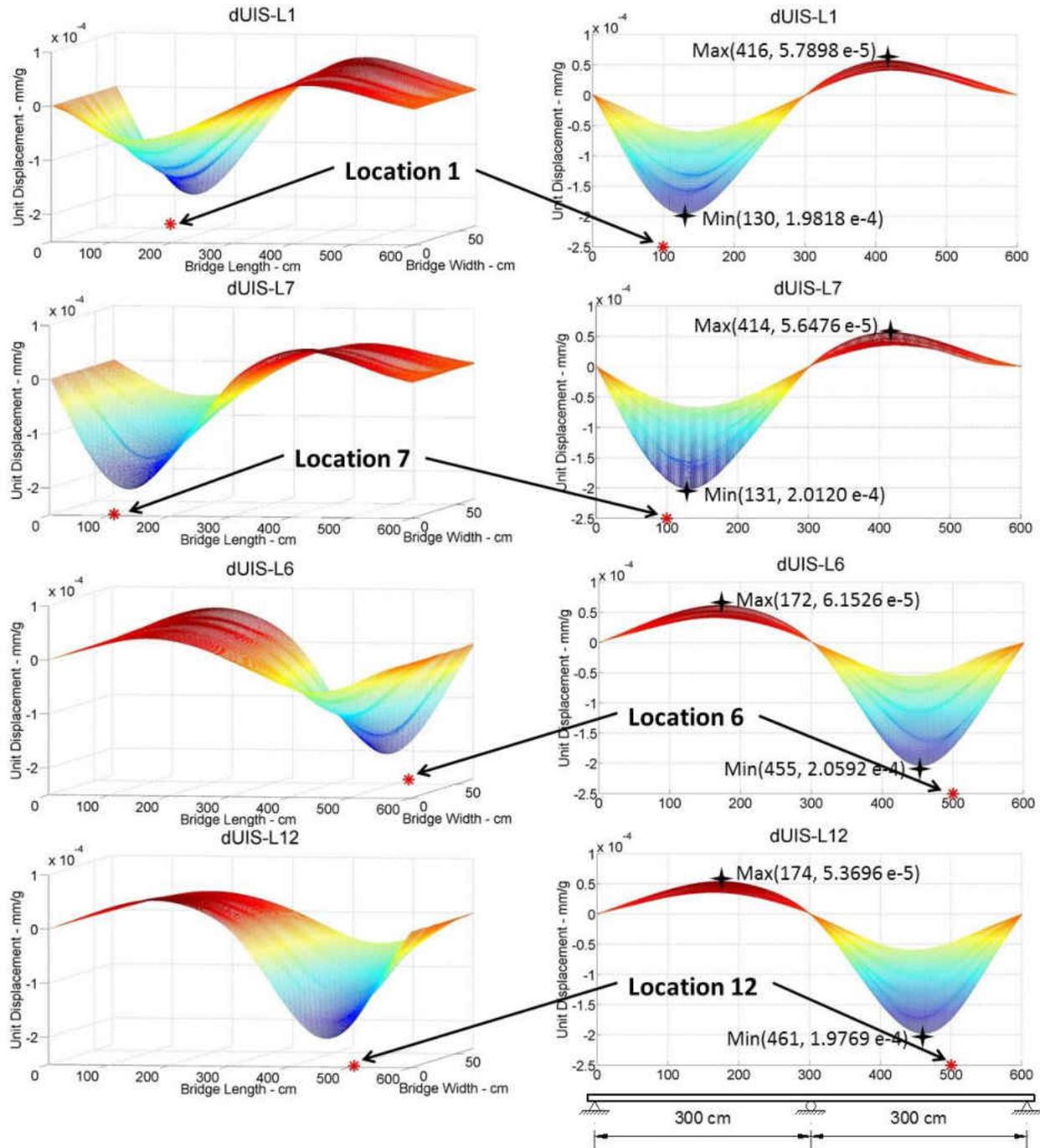


Figure 56: The displacement UIS's of the group *I* at the locations *L1*, *L6*, *L7*, and *L9*

Figure 56 shows a comparative illustration of the UIS's at the locations *L1*, *L6*, *L7*, and *L9* (group *I* of the measured positions) that are assigned symmetrically. Thus, these UIS's are expected to be similar in shape as well as the maxima values. It is observed that the shapes of the UIS's are totally analogous after flipping these about the symmetrical axis. The maxima points of the UIS's are also identified and then be detailed in Table 10 including their locations as well as their values. The information of those maxima values affirms that the obtained UIS's at the location *L1* and *L7*, symmetrically placed over the median axis of the bridge deck, are almost identical. This similar occurrence is also observed for the UIS results of the locations *L6* and *L12*. A little alteration is seen when comparing the UIS's from *L1* and *L7* to the UIS's from *L6* and *L12*. Although the highest errors of the maxima values are approximately 7.2%, the most important info like the minimum values yield the errors at a rate less than 2.6%.

In summary, an initial verification of the UIS's constructed from a particular *Set* (Figure 53) is deployed and depicted in Figure 55 and Figure 56. The evaluation illustrates that the proposed process for constructing UIS from direct measurements returns reliable and plausible outcomes, especially from structural analysis perspective. In the subsequent section, the UIS's obtained from the other *Sets* (a total of fifteen *Sets*) are be compared by means of statistical analysis to verify their consistency, which is the most important characteristics of a potential damage feature.

Table 10: Error evaluation for the group *I* of the measured symmetrical positions

	Distance from Max to the Closest End (cm)		Distance from Min to the Closest End (cm)		Max Value (unit disp.) mm/gram		Min Value (unit disp.) mm/gram	
	Value	Error	Value	Error	Value	Error	Value	Error
dUIS-L1	184	2.8%	130	-4.6%	5.7898E-05	0.9%	1.9818E-04	-1.3%
dUIS-L6	172	-3.9%	145	6.4%	6.1526E-05	7.2%	2.0592E-04	2.6%
dUIS-L7	186	3.9%	131	-3.9%	5.6476E-05	-1.6%	2.0120E-04	0.2%
dUIS-L12	174	-2.8%	139	2.0%	5.3696E-05	-6.5%	1.9769E-04	-1.5%
dUIS-average	179	0.0%	136	0.0%	5.7399E-05	0.0%	2.0075E-04	0.0%

As mentioned previously, a total number of fifteen (15) *Sets* are developed by combining ten (10) different tests out of forty (40) conducted for each *Set*. For every *Set*, the UIS's at all locations are constructed. Hence, the protocol produces fifteen (15) UIS's for each location. To verify the consistency characteristics of the obtained UIS's, some parameters of UIS such as the surface volume and maxima values (only the *Min Values* are interested herein) are calculated. For instance, the volume and maxima values of these UIS's at the location L1 for all *Sets* are depicted in Figure 57a as well as in Figure 57b respectively. The Figure shows that those outcomes are highly steady as they alter within a narrow range. A measure of data dispersion is also performed that yields the coefficient of variation (CV) value of 1.27% for the UIS volume data and the value of 1.00% for the UIS maximum data, which are greatly low. The error analyses implemented for other measured locations are illustrated in Table 11 and Table 12. The results of CV parameters for all cases illustrate that the maximum of CV value is 1.70% and many other CV values are less

than 1.00%. The evaluation confirms the successful consistency characteristics of the UIS's that are attained by the proposed framework.

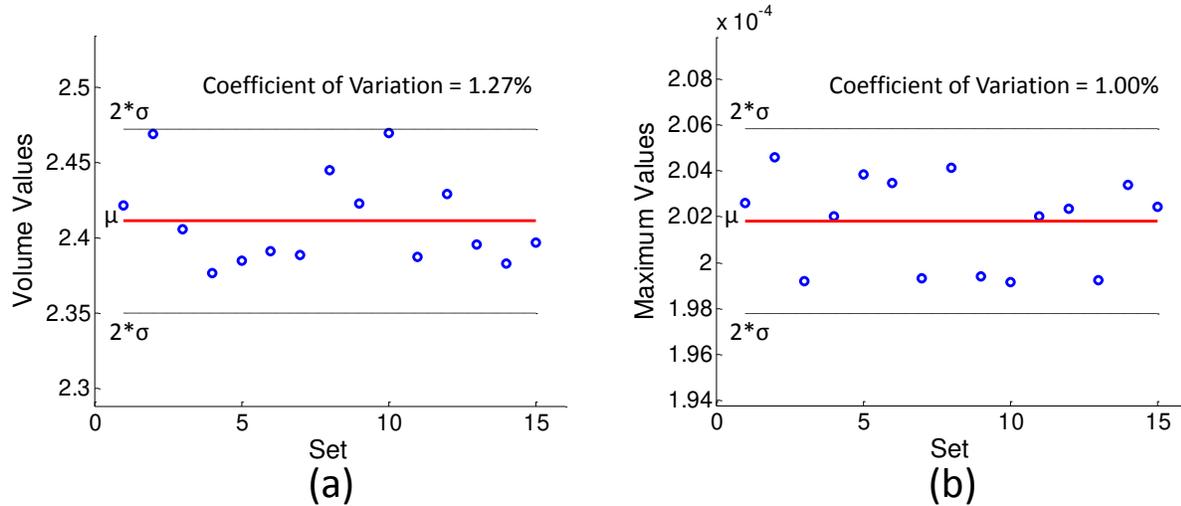


Figure 57: The parameter values of the UIS's at the location L1 for all Sets a) Volume of the UIS's; b) The maxima values of the UIS's

Table 11: The volume values of the UIS's for every Set

	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10	L11	L12
μ	2.411	2.678	2.417	2.403	2.741	2.402	2.421	2.756	2.537	2.548	2.711	2.321
σ	0.031	0.032	0.025	0.022	0.022	0.020	0.042	0.034	0.033	0.024	0.023	0.020
CV	1.27%	1.19%	1.04%	0.90%	0.79%	0.82%	1.74%	1.22%	1.28%	0.94%	0.85%	0.88%

Table 12: The maximum values of the UIS's for every Set

	L1 (10 ⁻⁴)	L2 (10 ⁻⁴)	L3 (10 ⁻⁴)	L4 (10 ⁻⁴)	L5 (10 ⁻⁴)	L6 (10 ⁻⁴)	L7 (10 ⁻⁴)	L8 (10 ⁻⁴)	L9 (10 ⁻⁴)	L10 (10 ⁻⁴)	L11 (10 ⁻⁴)	L12 (10 ⁻⁴)
μ	2.018	2.152	1.819	1.854	2.281	2.094	2.032	2.218	1.826	1.932	2.232	2.017
σ	0.020	0.013	0.009	0.025	0.025	0.026	0.028	0.027	0.019	0.008	0.008	0.007
CV	1.00%	0.59%	0.51%	1.32%	1.12%	1.26%	1.40%	1.21%	1.02%	0.42%	0.34%	0.35%

Conclusions

This chapter introduces a novel type of damage indicator involving structural parameters for Bridge Health Monitoring (BHM). The proposed damage indicator is Unit Influence Surface (UIS) that is a 3-D version of Unit Influence Line (UIL), another interest catcher damage indicator in BHM recently. The procedure for constructing a UIS requires series of measurements that are proposed previously in Chapter 2 and Chapter 3. Since the measurements introduced in this dissertation are all vision based, the UIS can be extracted by means of a non-sensor monitoring protocol that brings about a potent and convenient implementation alternative for real-life bridges. In brief, a UIS is constructed by deploying surface fitting algorithms (e.g. Bilinear Interpolation and Thin-plate Spline Interpolation) to a set of *UIS-points*, each of which describes a state of *vehicle loading-structural response* (input-output) interaction while traffic passes on the bridge deck. Hence, a UIS-point (X_j, Y_j, U_j) is developed from a *configuration location* (X_j, Y_j) derived from vehicle positions and a *normalized displacement* U_j at the measurement location. While *the configuration location* can be effortlessly calculated from the detected vehicle positions, *the normalized displacement* is determined by operating an iterative approximate procedure on coupled information of vehicle and structural displacement data.

The UIS's are extracted for the 4-Span Bridge in UCF Structural Lab. To validate the outcomes of the proposed protocol, a total of twelve (12) positions under main girder flanges are selected for constructing the UIS's. Since these positions are placed symmetrically over two symmetrical axes (middle-support axis and median of the deck), the UIS's at the symmetrical locations illustrate identical shapes and maxima values (error values from 0.2% to 7.2%). In addition, the UIS's determined from neighboring positions are observed complying with structural analysis perspective to supplement the UIS plausibility. The UIS consistency is also verified since

it is the most essential characteristics for a damage indicator. A consistent verification method is deployed by constructing numerous (15) UIS's for each measurement locations from a database of forty (40) tests. The error analyses show that the (15) UIS's at any location are identical. A measure of data dispersion for the UIS parameters (e.g. UIS volume and UIS maximum data) yields the maximum of CV (coefficient of variation) value to be 1.70% and many other CV values to be less than 1.00%. Quantitative analysis of the obtained UIS confirms the consistency and reliability. With these characteristics, the UIS promises a great potential to become a sensitive damage indicator. The UIS's constructed by the proposed method are exploited for damage detection as well as damage localization in the next Chapter.

CHAPTER FIVE: DETECTION AND LOCALIZATION OF DAMAGE USING UIS OBTAINED WITH LIMITED NUMBER OF SENSORS

Introduction

Structural Identification (St-Id), among others, is a broadly interdisciplinary research area in Structural Health Monitoring (SHM) involving various implementations such as structural-characteristics extraction, structural inspection, structural control, and also physics-based model updating for constructed systems (F. Catbas, Kijewski-Correa, & Aktan, 2011). Since the ultimate aim of a St-Id system is to prevent failures likely to occur during routine operation of constructed structures, damage identification is always an essential component that provides clues and proofs for decision making. Ideally, a comprehensive damage identification scheme should be developed in three steps including 1) damage detection, 2) damage localization, and 3) damage quantification, sequentially. Following that scheme, the first step involving damage detection is to predict problems as well as to provide information for the next steps. The second and third steps of interpretation aim to support the proofs of damage, which help inspectors and also project owners for better decision-making. In practice, damage qualification is commonly a very particular implementation since critical levels of damage are ranked differently for each specific constructed structure. Examples of these are the structure type, importance of the structure in the infrastructure network, age of the structure, and specific requirements of owners. In scope of this work, only the two first steps of a damage identification paradigm are studied including damage detection and localization for bridges.

Damage detection module of a St-Id system is developed as a sort of anomaly detection that yields either *damaged structure* or *non-damaged structure* states. In this type of anomaly detection, a damage condition is identified as an outlier due to the damage data creating an

unbalance within the usual nature of the whole measurement data sets. Design of such an anomaly detection necessitates damage indicators as exemplified in the previous chapters. The type of indicators can be selected from state-of-the-art features (e.g. natural frequencies, damping ratios, mode shapes etc.), or novel ones can be proposed (such the UIS damage indicators herein). The selected damage indicators are then extracted from measurement data of the monitored structure for damage detection purposes. As a requirement in data mining, long-term measurement data is demanded for developing any damage detection paradigm. Although numerous damage detection practices have been conducted, those can be categorized based on the utilized anomaly detection techniques as unsupervised and supervised methods.

Unsupervised anomaly detection technique is to construct a binary classifier from only given labeled positive dataset. Hence, the technique for damage detection is developed just from long-term measurement data that is acquired while the monitored system is in healthy condition. Due to damage indicators demonstrating a steady level for a certain structural condition (as mentioned in Chapter 4), a threshold pointing the healthy case for those indicators can be developed. Once the confidence threshold is achieved, the input condition of the monitored structure is identified by comparing the damage indicator to the threshold. If that input damage indicator is ranked as an outlier (out of the threshold range), the structure is perceived as in damage condition, and vice versa. Since the unsupervised anomaly detection approach does not demand data of a damaged state, it provides advantages especially for monitoring real-life structures, where data for damage scenarios is unavailable in many cases. Sample studies of unsupervised damage detection in St-Id employ variety of damage indicators such as natural frequencies (Nandwana & Maiti, 1997; Salawu, 1997), mode shapes (Hu & Afzal, 2006; Lee, Lee, Yi, Yun, & Jung, 2005), modal flexibility (F Necati Catbas, Brown, & Aktan, 2006; F. N. Catbas, M. Gul, & J. L. Burkett,

2008), AR vector (Gul & Catbas, 2011; Q. Zhang, 2007), statistical coefficient (F. N. Catbas et al., 2012; M Malekzadeh, Atia, & Catbas, 2015), and UIL (OBrien et al., 2015; R. Zaurin & Catbas, 2010),.

Although employing unsupervised anomaly detection technique for identifying structural damage has advantages, the method also discloses a shortcoming involving false positive alarm due to other influences besides real deterioration. Those influences may be caused by noisy measurement data, sensor malfunction, environmental effects (e.g. temperature, wind). In addition, establishing reliable thresholds for unsupervised anomaly detection is subjective and challenging when there is lack of understanding about structural behaviors at damage conditions. To mitigate these issues, given data sets from damage scenarios should be introduced. Once negative (damage) data sets are available, the damage indicators obtained from that data can be used to verify the confidence and reliability of the thresholds. Thus, the supervised anomaly detection technique can be implemented to yield a more robust damage detection module that might even be capable of obtaining damage types or damage locations. Several supervised learning algorithms are utilized for structural damage recognition including Fuzzy Neural Network (Gul, Catbas, & Hattori, 2013), Artificial Neural Network (Arangio & Beck, 2012; Kao & Loh, 2013; Shu, Zhang, Gonzalez, & Karoumi, 2013), and Support Vector Machines (Farrar & Worden, 2012; Magalhães, Cunha, & Caetano, 2012). A challenge when conducting the supervised technique is how to acquire damage data sets. In laboratory monitoring studies, damage data can be conveniently obtained by simulating damage scenarios. However, damage reproduction in a real-life structure is a very difficult task, and commonly be rejected by the owner of the structure. Anyway, several real-life SHM implementations have been conducted while some damage schemes were being produced in

bridges for short time such as (F. Necati Catbas, Mustafa Gul, & Jason L. Burkett, 2008; Gul et al., 2013; Ricardo Zaurin, Khuc, & Catbas, 2016).

Damage localization is the second step of damage identification. After the damage detection module gives the alarm, deteriorated locations are considered for further inspection by sending inspectors to the damage region or terminating operation on the structure. Locating damage in big constructed systems (e.g. bridges, stadiums, pipelines) especially helps save time and labor-work for detailed inspection. In general, damage locations are pinpointed based on the positions of sensors in a network. The insight of this idea is acquired from the observation involving the shifting magnitude of the damage indicator determined from different sensors when damage occurs. For instance, the damage feature values obtained from a particular sensor near to a deteriorated position are altered more significantly than the ones extracted from remaining sensors, which are away from the damage location. Thus, a sensor network is better to be deployed densely and at critical regions of the structures in SHM studies. Subsequently, the damage indicator values from all sensors are obtained and then quantified following the diagnosis of the damage location by looking at the neighboring sensors that show the highest changing of damage indicator level. The framework to determine a damage location based on the reference positions of sensors is introduced in a great number of literature studies (F. N. Catbas et al., 2012; Gul & Catbas, 2011; Lee et al., 2005; Orcesi & Frangopol, 2010; R. Zaurin & Catbas, 2010). Recently, damage localization can be also achieved by means of machine learning approach. This method categorizes the healthy status and all damage scenarios that may occur at different positions of a structure as various groups. As an example, a baseline case and four (4) damage scenarios yield five (5) structural condition groups. Subsequently, a classifier is developed from the training data sets acquired from the baseline case and also damage scenarios. Finally, an input data measured

from the structure is introduced to the classifier that points out the structural status. This approach is introduced in (Masoud Malekzadeh, 2014) and is successfully conducted with limited damage locations.

Motivation and Objectives

Damage identification is the ultimate aim of SHM studies that provides early notification for profound inspection on constructed systems. An impeccable damage identification framework comprises three steps of evaluation namely 1) damage detection, 2) damage localization, and 3) damage qualification. However, in academia, damage qualification is of less interest for the reason that qualification criteria are biased for a specific structure. Thus, only damage detection and localization schemes are introduced by means of the novel damage indicators that is proposed in Chapter 4 of this dissertation.

In general, damage detection is about defining two likelihoods that are *damaged structure* or *non-damaged structure* by making use of binary classifiers. In regards to this, damage detection is commonly developed by employing anomaly detection techniques categorized as unsupervised and supervised learning. It is seen that the unsupervised damage detection is more popular and can be implemented in most of SHM studies. This is due to the fact that the technique only requires the measurement data obtained from the sound structural condition (positive data sets) for training and then determining damage thresholds. On the other hand, the supervised damage detection demands additional measurement data of a damaged state that is to serve as negative data sets for learning task. Obviously, supplementary damage data assists the supervised damage detection to have more robust classifiers, which, in return, yield more confident and reliable thresholds along with damage types or damage locations. The main shortcoming of the supervised technique is the

difficulty of collecting damage data. Although it is convenient and easy to simulate damage scenarios for a particular structure with modular parts in laboratory, it is extremely challenging and even impossible to do soon a real-life constructed system due to numerous legitimate requirements and structural safety concerns.

The next step of damage evaluation, damage localization, is also introduced in many studies due to its benefits, especially for complex and big constructed systems. Based on the observation that the higher alteration of damage indicator is acquired from the sensors in the vicinity of damaged regions, most of damage localization techniques are developed by quantifying changes in damage indicator obtained from a network of sensors. Hence, the location of the sensor corresponding to the highest alteration of damage indicator is hypothesized as the location of damage. Although the approach can achieve the goal, it requires a dense array of sensors that is costly for a real-life constructed system.

The first part of implementation in this Chapter is to develop a damage identification framework based on the conventional approach. In a few words, a damage detection module is introduced by employing a new damage indicator, the Unit Influence Surface (UIS), which is described in Chapter 4. Since the UIS is not represented as a function of single parameter, there is an intermediate step to implementation to reduce that high dimensional space by calculating the volume of the UIS. The calculated volumes are then used for anomaly detection by establishing confident damage thresholds. Damage locations are also determined by comparing the normalized changes of the volume levels over the sensor network. This part aims to evaluate capability of the new damage indicator (UIS) using classical approach for identifying damage in bridges.

The second part of this Chapter is to introduce a new damage identification approach by exploiting the high dimensional space of the UIS. Please note that the UIS comprises finite vertical

ordinates that correspond to the unit loading locations. Since those finite vertical ordinates can be extracted at any locations along a bridge deck, a damage occurrence and its location can be discovered by quantifying these. Moreover, since the proposed method does not require a network of sensor, it enables the possibility of a limited sensor SHM framework.

The details of the theoretical background are described in next sections. The accuracy of the framework is verified on a small-scale bridge in the UCF Structural Lab, where several damage scenarios are simulated for damage identification purposes.

Theoretical Background

Classical Approach: Damage Identification based on a Network of Measurement Positions

The damage identification method based on the conventional approach using the new damage indicator (UIS) consists of two phases. The first phase is to establish thresholds for the healthy structural scenario (so-called *Baseline*) as illustrated in Figure 58. In brief, a series of UIS's is constructed for the *Baseline* case following the framework described in Chapter 4. Due to the UIS being a fitted surface function, it is discretized and then represented in matrix form for convenient analysis purposes. Subsequently, the *Baseline* UIS-volumes are calculated (Equation 67) to develop the *Baseline* distribution. Thresholds are then defined equal to ± 2 times of the standard deviation away from the mean value of *Baseline* distribution (corresponding to a confidence level of 95%) as illustrated in Equation 68.

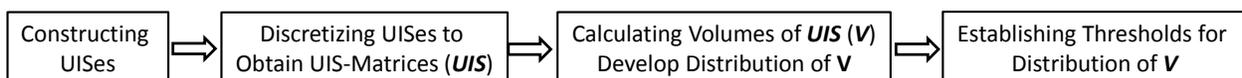


Figure 58: Establishing thresholds for the healthy structural scenario (*Baseline*)

In the second phase, four damage scenarios (named *Damage 1, 2, 3, and 4*) are simulated on the UCF 4–Span Bridge. The *Damage* UIS’s are also obtained and discretized for determining *Damage* UIS-volumes in the same manner as shown in Figure 58 (from step 1 to 3 only). Those *Damage* UIS-volumes are employed for verifying or even modifying (if needed) the thresholds, which are developed in the first phase.

For damage localization objective, a *damage weight* that measures the change of UIS-volumes in percentages between *Baseline* and *Damage* scenarios is introduced. These *damage weights* are determined for every sensor; and then these values are concatenated to get the highest weight. The position of the sensor corresponding to the highest weight is a neighboring damage position.

The UIS Discretion and Matrix Form of the UIS

As mentioned previously, the UIS is a fitted surface function $S(X, Y)$, where X stands for the coordinate of the bridge-width axis and Y of the bridge-length axis. To make the UIS analysis more convenient, the UIS surface is transformed into matrix form by discretizing by means of a preset grid. Since the obtained UIS size is 80 x 600 cm (ref. Chapter 4), the grid size of 80 x 600 cells is selected for the UIS discretion. Hence, dimensions of a cell are 1 x 1 cm; and the vertical ordinates at every node of the grid are calculated from the fitted surface function $S(X, Y)$. By that, a particular form of the UIS now is represented as a matrix with 81 rows and 601 columns denoted as $UIS_{81 \times 601}$.

Volume of the UIS

Once the UIS is discretized, its volume can be effortlessly calculated as follows:

$$V = \sum_{i=1}^{81} \sum_{j=1}^{601} U_{i,j} \quad (67)$$

Where $U_{i,j}$ is a vertical discretized ordinate of the UIS at location (X, Y) of $(i-1 \text{ cm}, j-1 \text{ cm})$. This value is an element in the matrix $UIS_{81 \times 601}$ at *row*- i and *column*- j .

Thresholds of the UIS-Volume (V)

$$Th = \pm 2\sigma + \mu \quad (68)$$

Where σ is the standard deviation of the V distribution; and μ is the mean of the V distribution.

Detecting and Localizing Bridge Damage Simultaneously based on UIS's Obtained from Limited Measurement Locations

In general, a damage indicator solely demonstrates structural behaviors with its capability of tracking structural alteration so that deterioration on constructed systems could be directly detected. On the contrary, the vicinities of damage are pinpointed indirectly via sensor locations thereby requiring a dense array of sensors which can be of high cost and complicated to be deployed. In this dissertation, it is fortunately observed that the UIS consists of not only structural responses (represented by vertical ordinates U) but also positions of these responses (represented by coordinate X and Y). That insight empowers the idea to utilize UIS's for both objectives: damage detection and damage localization without the need for a sensor network. Success of such a framework with limited number of sensors is a promising progress involving improvement of a fast, low-cost and portable SHM system for a broad adoption. .

The UIS based damage identification is proposed by quantifying changes of every cell on the first derivative of the UIS matrix on the Y axis (the bridge-length axis), named as the UIS- Y matrix. In a few words, this new framework is a combination of finite classifiers that are developed

for every element of the indicator matrix (the UIS-Y matrix herein). For example, at each cell (*row-i, column-j*) of series of matrices UIS_Y, a distribution of cell-intensity values is created, called the *Baseline-Cell* distribution. Subsequently, thresholds, which are called the *Cell thresholds*, are defined for each *Baseline-Cell* distribution that equal to ± 2 times of the standard deviation away from the mean. Finally, *threshold matrices* are established from the *Cell thresholds* as depicted in Figure 59.

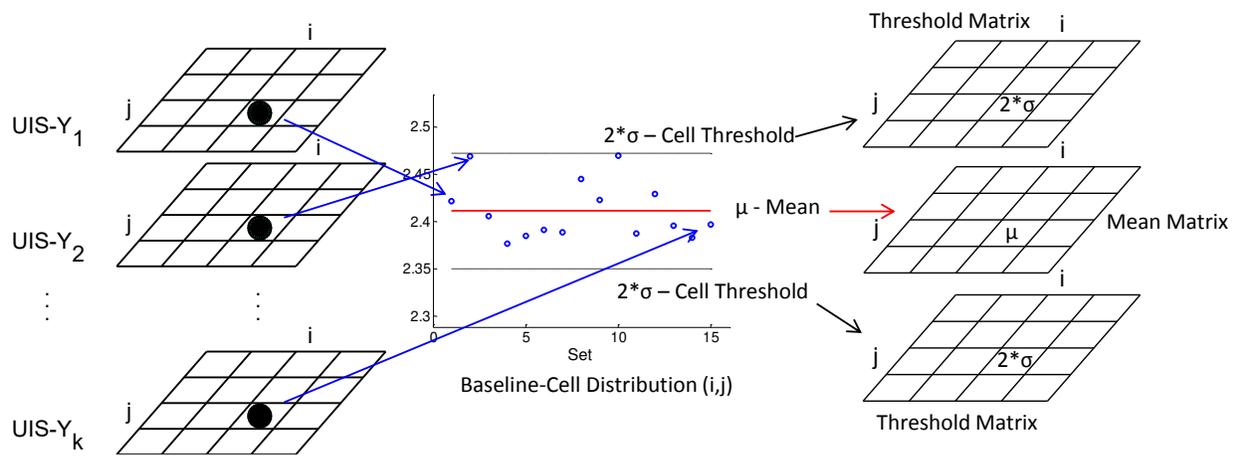


Figure 59: Establishing thresholds for the new UIS based damage identification

Once the *threshold matrices* for the healthy scenario (*Baseline*) are obtained, the input matrix UIS_Y is compared with the *threshold matrices* for damage identification. A comparison based on the *Cell* threshold is made for each cell to classify if the cell is either damaged or non-damaged. The overall structural status corresponding to the given UIS-Y is ranked as damaged if the number of damage-cells is higher than a preset level, and vice versa. The change in each cell is also calculated. Those changes are then quantified to obtain the region of cells altering the most. Hence, that region is decided as the vicinity of damage location.

The First Derivative of the UIS matrix

The parameter deployed for developing the new UIS based damage identification is the first derivative of the UIS on the Y-axis (the bridge-length axis). Since the UIS obtained herein is the displacement UIS, its first derivative is the slope (tilt) UIS. Due to the inability of measuring displacements of structures at supports, using slope (tilt) is expected to address the challenge of damage identification at (or near) supports. The first derivative of the UIS matrix can be determined by convolving the derivative kernel k_y on the matrix $UIS_{81 \times 601}$ as follows:

$$UIS_Y = k_y \otimes UIS$$
$$\text{where } k_y = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad (69)$$

Where UIS_Y is the first derivative of the matrix UIS ; and \otimes is a mathematical operator called convolution applied to two matrices in this certain case.

Laboratory Verification

Simulation of Damage Scenarios on UCF 4-Span Bridge

Since the UCF 4-Span Bridge is an SHM apparatus in UCF Structural Lab, it is designed in such a way that a number of common damage scenarios encountered in bridges can be simulated. The scenarios simulated in the 4-Span model are typical damages (global and local) that are found as a result of comprehensive investigations carried out by SHM research group in UCF and with the help from Florida Department of Transportation engineers (F Necati Catbas et al., 2010; R. Zaurin & Catbas, 2010). Global damage is the deterioration related to changes in boundary conditions such as corrosion, a roller support being stuck, and support settlement (or movement). This type of damage causes internal-force redistribution and also affects other elements

extensively. On the other hand, local damages are signs of less critical deterioration emerging as loosened bolts, local section loss (or degradation) due to corrosion-penetration or shallow cracks, or even lateral bracket failure. In this study, both global and local damages are reproduced to verify the proposed damage identification framework and also the UIS damage indicator reliability. The global damage scenarios are simulated by converting the pinned supports of the 4-Span Bridge to the fixed supports whereas the local damage scenarios are conducted by loosening several bolts that connects the steel deck sheet with the main girders. The simulated global and local damages are illustrated in Figure 60.

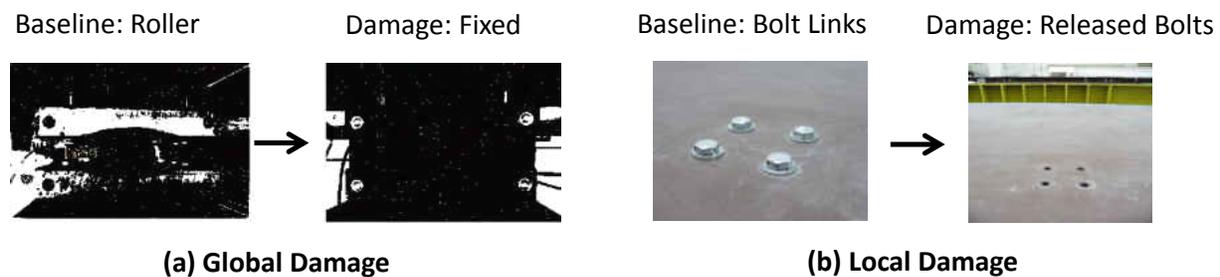


Figure 60: Damage simulation in the UCF 4-Span Bridge; a) Global damage by altering boundary condition at supports; b) Local damage by releasing composited connections

A total number of four damage scenarios is simulated including two global (*Damage 1* & *Damage 2*) and two local (*Damage 3* & *Damage 4*). The *Baseline* and *Damage* scenarios are detailed as follows and also depicted in Figure 61.

- ✓ Baseline: The Baseline is the healthy condition of the UCF 4-Span Bridge. It is described as in Figure 61a.
- ✓ Damage 1: The global *Damage 1* is simulated by converting one out of the two roller supports at the center to the fixed support. (Figure 61b).

- ✓ Damage 2: The *Damage 2* is also replicated as another global deterioration with two stuck supports: one at the center and another at end of the bridge (Figure 61c). The same way where the conversion of the pinned supports to the fixed as in *Damage 1* is followed.
- ✓ Damage 3: The *Damage 3* is a local damage case. The damage is simulated by releasing a connection between the steel deck sheet and the main girder of the bridge. Since the bridge model is designed as a modular system, releasing a composite connection results in a decrease of structural stiffness, especially at the neighboring area around the released connection. The position of the released connection is shown in Figure 61d.
- ✓ Damage 4: Another local damage scenario is reproduced herein. The composite connections are released by loosening the bolts. The damage locations are selected on both spans of the UCF 4-Span Bridge as illustrated in Figure 61e.

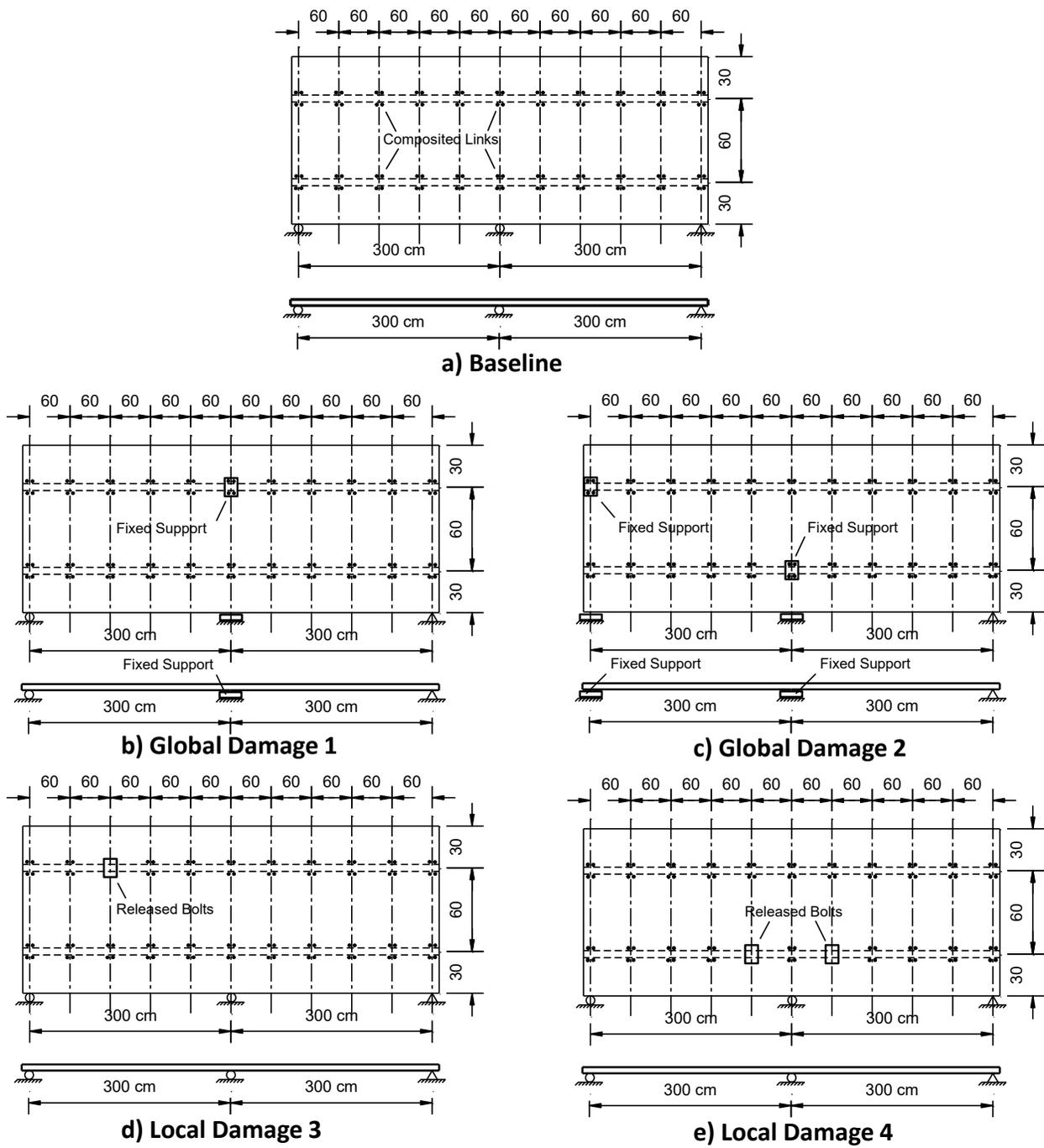


Figure 61: *Baseline* and *Damage* scenarios simulated in the UCF 4-Span Bridge

Damage Detection and Localization based on a Network of Measurement Locations

Baseline Results and Thresholds for the Healthy Condition of the Bridge Model

As mentioned previously in Chapter 4, a total of fifteen (15) *Sets* of data are acquired for calculating UIS's in the bridge model. For every *Set*, the UIS's at all (12) measurement locations are constructed so as to have fifteen (15) UIS's for each location. The volumes of (15) UIS's for all (12) measurement locations are calculated in terms of Equation 67 that are shown in Figure 63. Finally, thresholds which are calculated by Equation 68 are assigned to be equal to ± 2 times of the standard deviation away from the mean value of those volume distributions. Measurement locations and their categorization groups are re-illustrated in Figure 62.

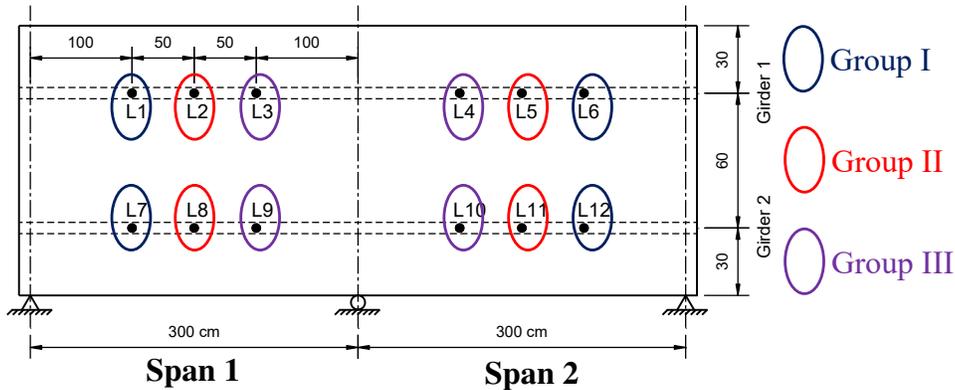


Figure 62: The measurement locations (L1 to L12), and their groups (Group I, II, III)

Damage Detection

Similar to the *Baseline* scenario, fifteen (15) *Sets* of data are acquired for each *Damage* scenario to simulate monitoring of the structure with damage. For each *Set* of data, twelve (12) UIS's are constructed corresponding to twelve (12) locations from L1 to L12. The volumes of those UIS's are calculated, and then are compared with the thresholds established in the *Baseline* case. If the volume values are outside the range of thresholds, the particular *Set* of data is classified as belonging to a damage case, and vice versa.

Figure 64 shows the results of damage detection for the case of *Damage 1*. It is seen that the indicators obtained from the damaged structure (red dots) are clearly shifted out of the threshold range (dash-black lines). These alterations are observed at all twelve (12) measurement locations that point out obvious and perhaps critical damage occurrence. A much clearer observation is captured in Figure 65 for the *Damage 2* when the data is processed and presented. The changes of damage indicators (UIS-volumes) are higher than the ones illustrated in Figure 64, notifying a more critical damage happening in this case. The remarks match well with the situations for both damage scenarios in which one deterioration is simulated in *Damage 1* while two are in *Damage 2*. The results also confirm that the new damage indicator (UIS) can effortlessly detect global damage, which is commonly more critical than local damage.

The results of the other damage scenarios are illustrated in Figure 66 and Figure 67 for the scenarios of *Damage 3* and *Damage 4*, respectively. Since these two damage scenarios represent local damages, the alterations of the UIS damage indicators are quite small. As seen in Figure 66, the UIS-volume indicators (red dots) are almost inside the range of thresholds although the change of mean values between the *Damage 3* and the *Baseline* conditions can be captured. It means that the *Damage 3* is hardly detected if using the preset thresholds calculated by Equation 68. Therefore, the modified threshold levels are required, for example, ± 1 time of the standard deviation away from the mean value. The damage detection results seem more successful for the *Damage 4* scenario. Figure 67 shows that the damage indicators are shifted out of the threshold range for eleven (11) out of twelve (12) locations. Even though the changes in this case are less than the ones observed in the *Damage 1* and 2, there is enough proof to decide on if there is a structural incident in the *Damage 4* case. Thus, the outcomes herein affirm that the new UIS damage indicators can detect not only global but also local damage, which is a minor type of

damage and is difficult to identify in many cases of SHM. In the next step, the UIS damage indicators are employed for damage localization by quantifying the changes of the UIS-volumes for every measurement position.

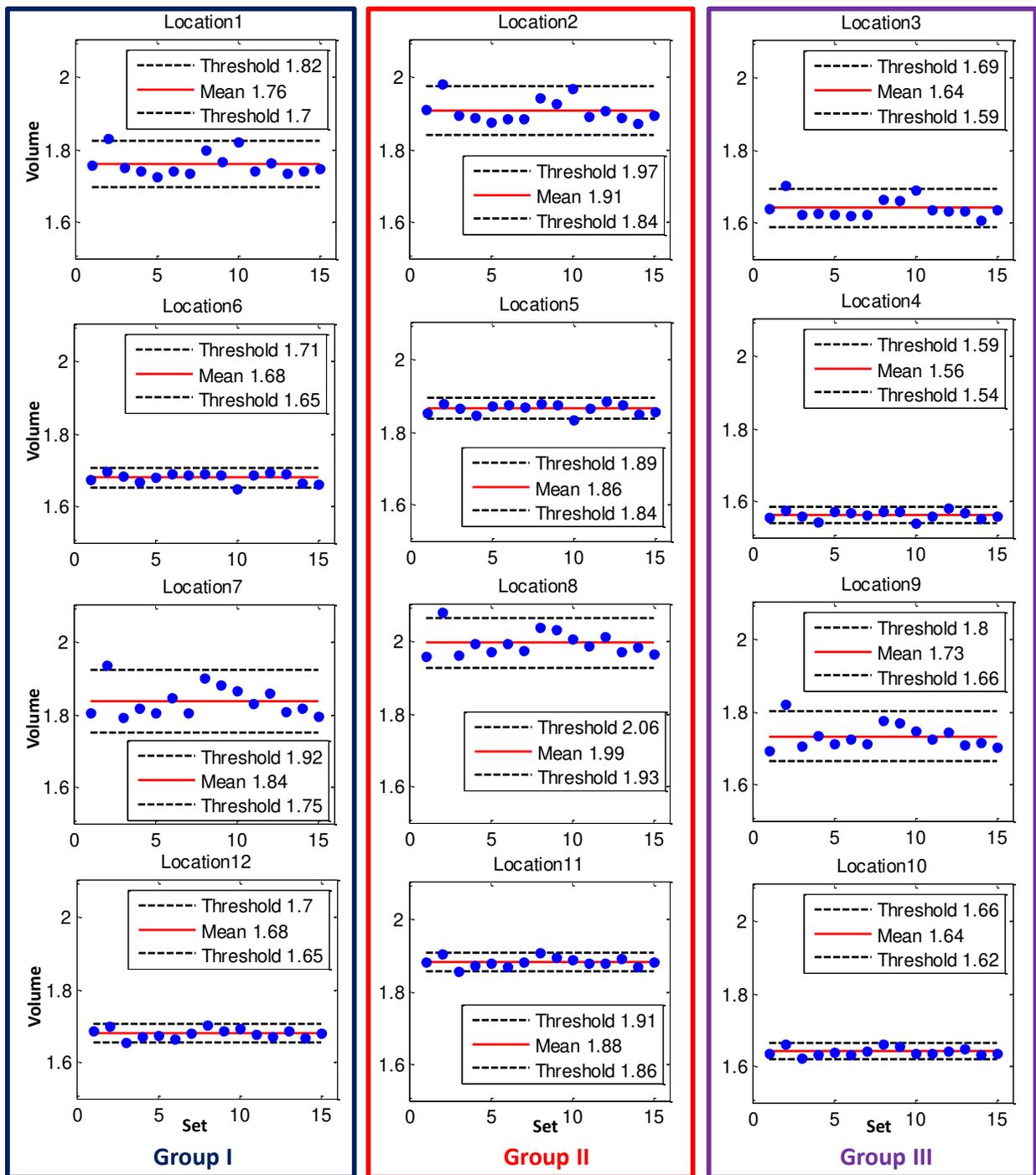


Figure 63: Volume of UIS's for Baseline scenario: results and thresholds

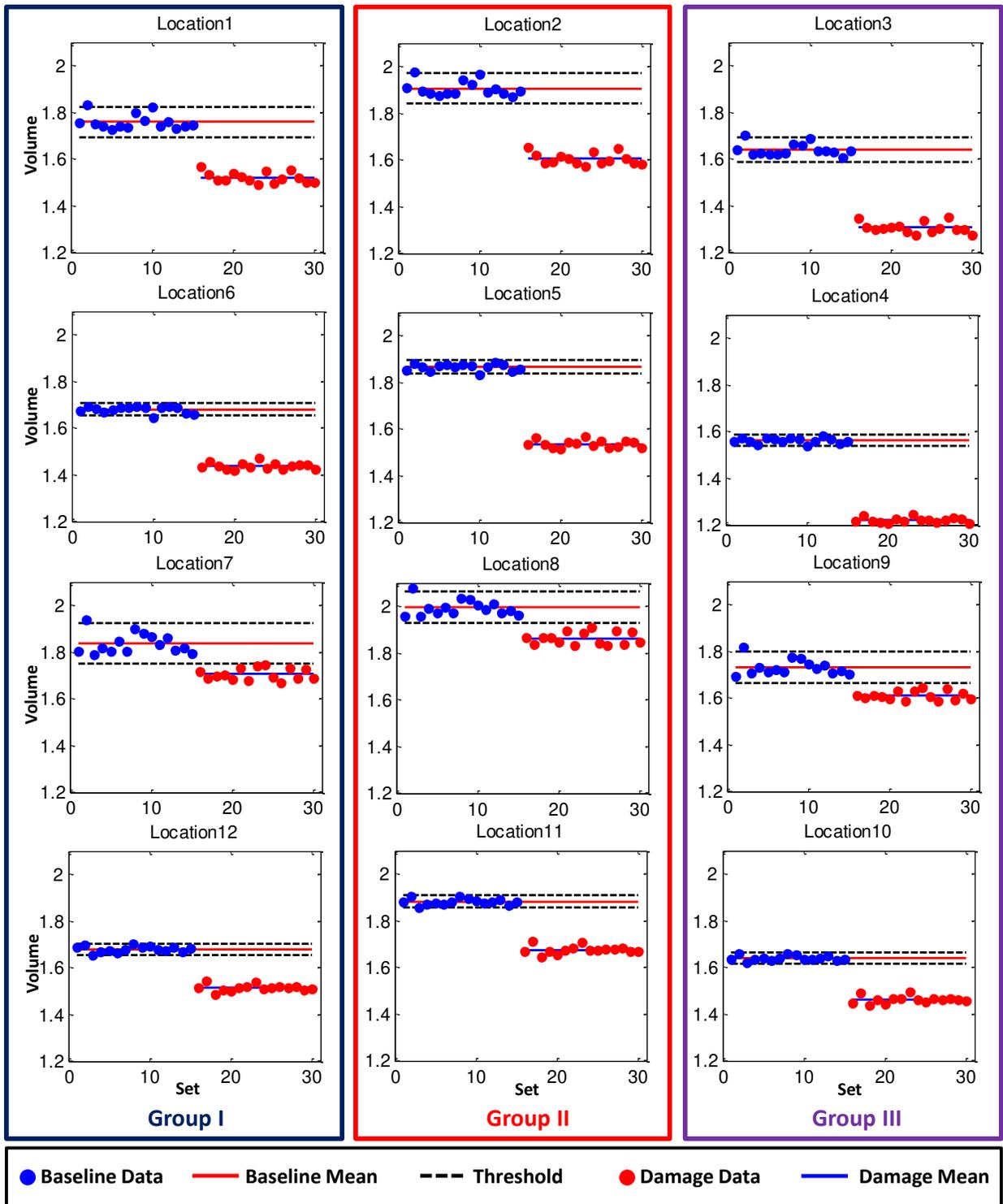


Figure 64: Damage detection for *Damage 1* scenario

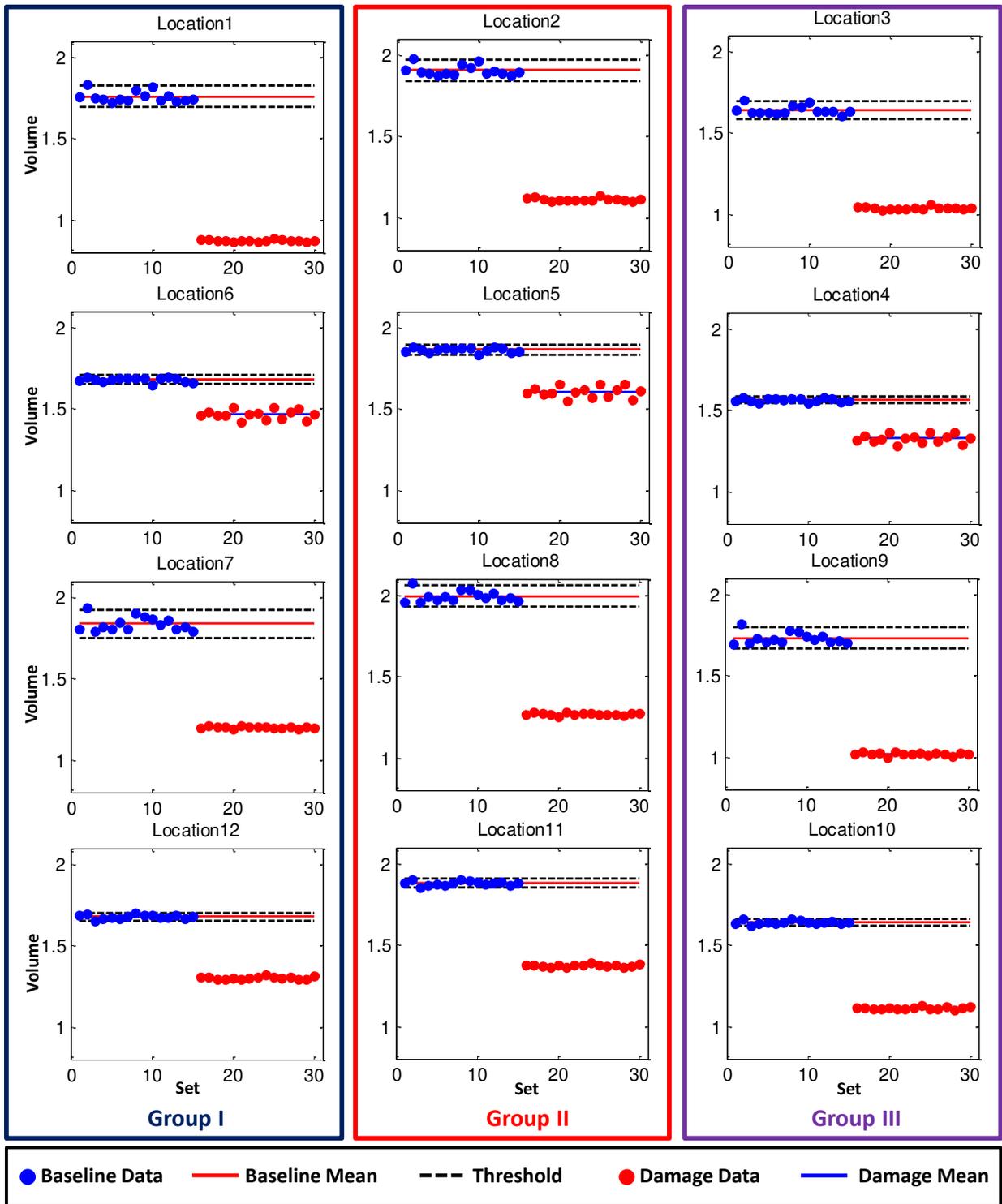


Figure 65: Damage detection for *Damage 2* scenario

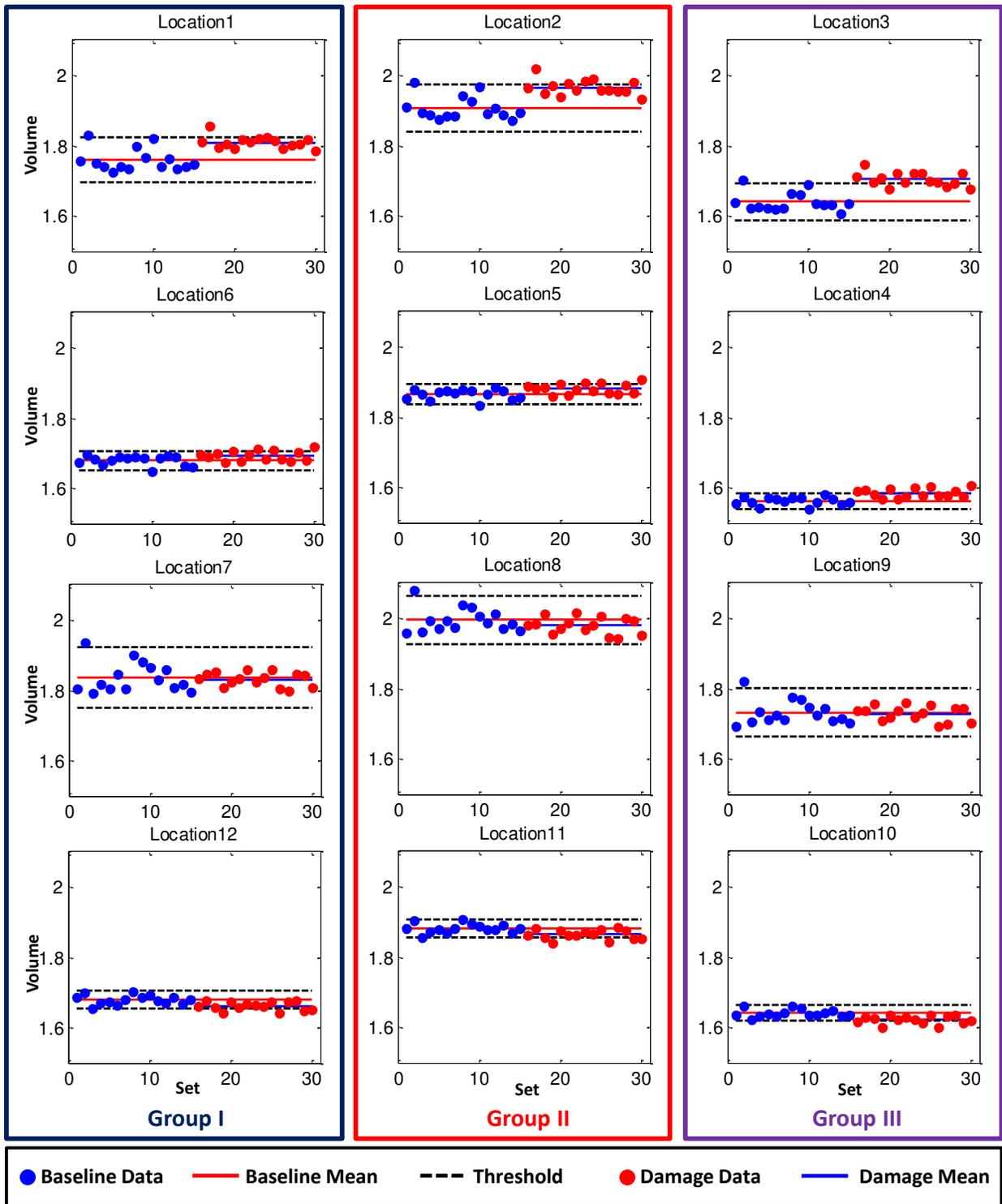


Figure 66: Damage detection for *Damage 3* scenario

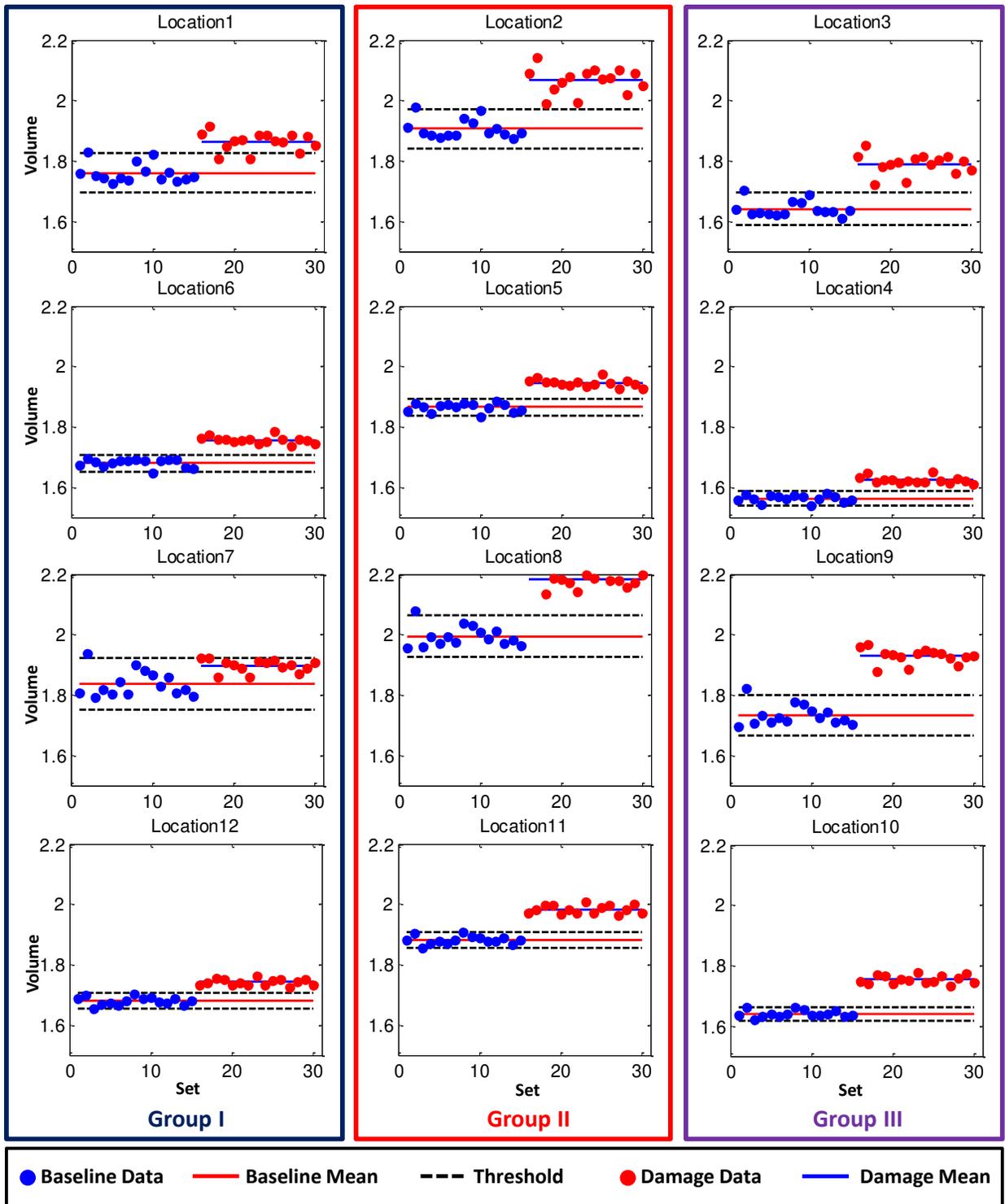


Figure 67: Damage detection for *Damage 4* scenario

Damage Localization

Damage localization is the second step of a comprehensive damage identification framework. That sort of implementation is achieved by ranking the changes of damage indicators on every sensor; and then by identifying the sensors matching with responses that alter the most. Thus, the damage positions are determined in the vicinity of these sensors. In this study, a *damage weight* is introduced as being a sort of standard for measuring the percentage changing of the UIS-volumes between the *Baseline* and the *Damage* scenarios. The *damage weights* (*DW*) are calculated as follows:

$$DW = \frac{\left| \frac{1}{n} \sum_{i=1}^n V_i - \frac{1}{m} \sum_{j=1}^m V_j^D \right|}{\frac{1}{n} \sum_{i=1}^n V_i} 100\% \quad (70)$$

Where V and V^D are the UIS-volumes calculated for the *Baseline* and the *Damage* scenario, respectively; n and m are the number of data *Sets* that are conducted for both the *Baseline* and the *Damage* cases (e.g. fifteen (15) *Sets* for each scenarios herein).

The *DW* values determined for the *Damage 1* scenario are presented with color-bar graph in Figure 68. It is observed that the highest *DW* values appear at the locations *L3* and *L4* (e.g. 20.4% and 21.8%, respectively) meaning that the damage/deterioration/change is on somewhere between the positions *L3* and *L4*. This finding matches with the damage location, which is on the center support that is simulated in the *Damage 1*. Similar behavior is obtained for the case *Damage 2* since the utmost changes of the UIS-volumes occur at the locations *L1*, *L9*, and *L10*, which are the neighbors of the reproduced damage areas (Figure 69). So, the analysis outcomes clearly confirm the existence of damages as well as their locations for the global damage cases (*Damage 1* and *Damage 2*).

The local damage cases are analyzed and their results are illustrated in Figure 70 and Figure 71 corresponding to the *Damage 3* and the *Damage 4*. Although the *DW* values in Figure 70 show a minor change in the UIS-volumes from the healthy to the damaged status (the maximum *DW* value of 3.9%), these values at the locations L1, L2, and L3 are exceeded about 2 to 10 times compared to the ones from the remaining positions. The indication suggests a possible deterioration, which matches with the damage location simulated in the *Damage 3*, along *Girder 2* of *Span 1* on the 4-Span bridge. The *DW* values for the *Damage 4* scenario are presented in Figure 71. It is also seen that the changes of the UIS-volumes are more noticeable with the *DW* values from 3.2% to 11.3%. Again, the utmost values of *DW* are on the locations *L3*, *L8*, *L9*, and *L10* that are around the simulated damage locations. So, the results herein confirm the capability of using the new UIS damage indication for not only detecting local damages but also localizing them. It is a very good sign since detecting and localizing that sort of local damage are always challenges in SHM implementations.

The framework employing the new damage indicator UIS is successfully verified in this section for both goals: damage detection and damage localization. The presented results show that the new UIS indicators can effortlessly identify the global damage. The indicators are also effective as to local damages despite leaving fuzzy clues and proofs behind. This remark is understandable and is experienced in most of the other state-of-the-art damage indicators for local damage is commonly capable of altering the structural characteristics slightly, which returns minor changes in structural responses. The main disadvantage of this conventional approach is that an array of sensors is required for damage localization. Hence, an alternative approach is proposed by selecting a limited number of measurement locations (sensors) for damage identification; and its verification results are described in the subsequent section.

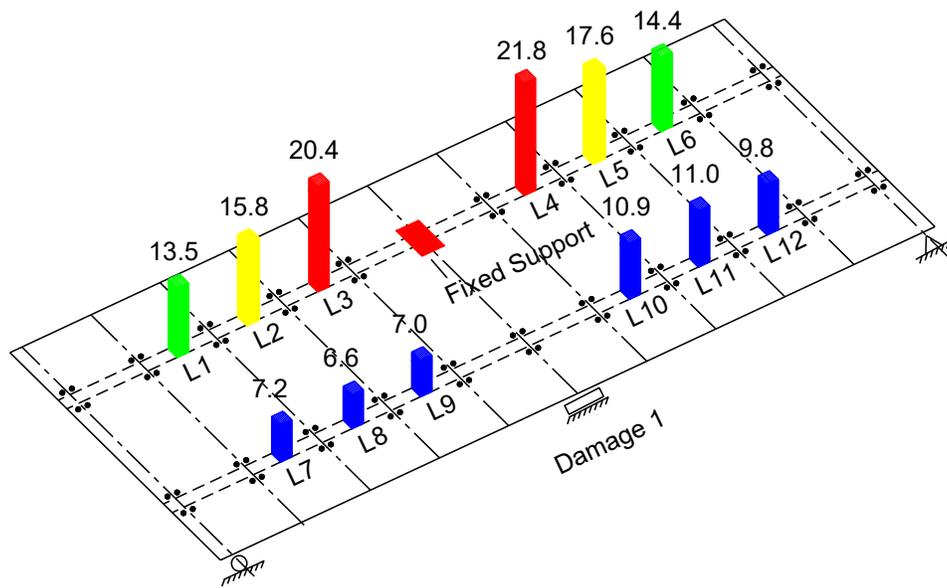


Figure 68: Damage weight DW for *Damage 1* scenario

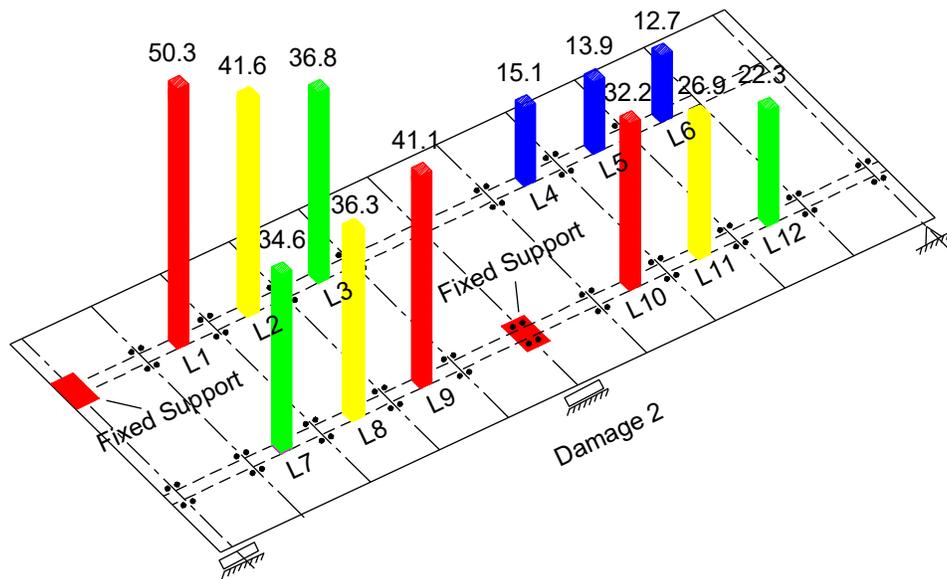


Figure 69: Damage weight DW for *Damage 2* scenario

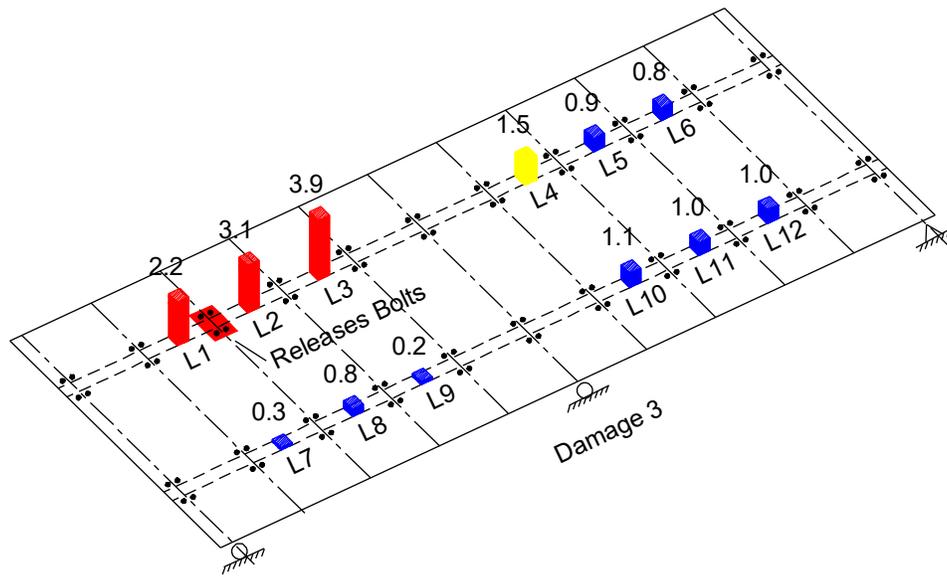


Figure 70: Damage weight DW for *Damage 3* scenario

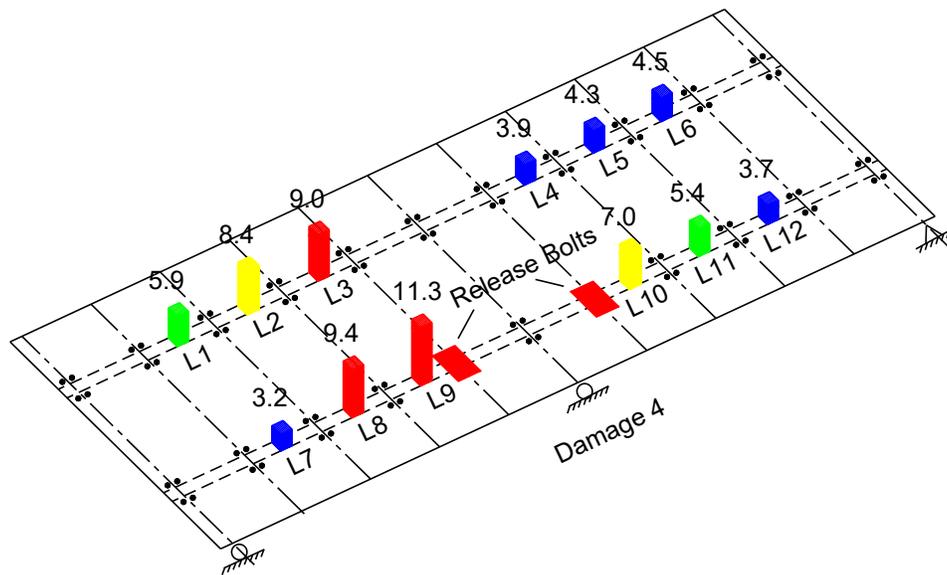


Figure 71: Damage weight DW for *Damage 4* scenario

Detecting and Localizing Damage in Bridges Based on Limited Measurement Locations

Establishing the Threshold Matrices for the Baseline Status

As detailed in the theoretical section, the *threshold matrices* comprising *upper-bound* and *lower-bound* matrices are developed based on the mean (M) value and the standard deviation (SDT) matrices of series of UIS-Y matrices. Since a UIS is the form of a matrix, its first derivative matrix UIS-Y is determined following Equation 69. A particular UIS-Y matrix is depicted in Figure 72. As fifteen (15) UIS's for each measurement locations are constructed, there is a distribution of fifteen (15) UIS-Y matrices; as well as the M matrix and the SDT matrix that is to be calculated. Finally, the threshold matrices (TH) are determined using Equation 71. The M (mean), the SDT (standard deviation), the *upper-bound*, and the *lower-bound* matrices for the *Baseline* case are illustrated in Figure 73.

$$TH = \pm 2 * SDT + M \quad (71)$$

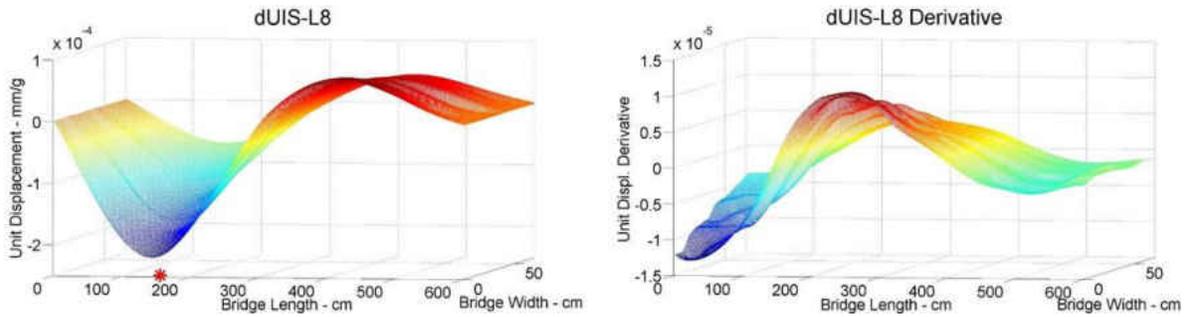


Figure 72: The UIS matrix and its first derivative UIS-Y matrix

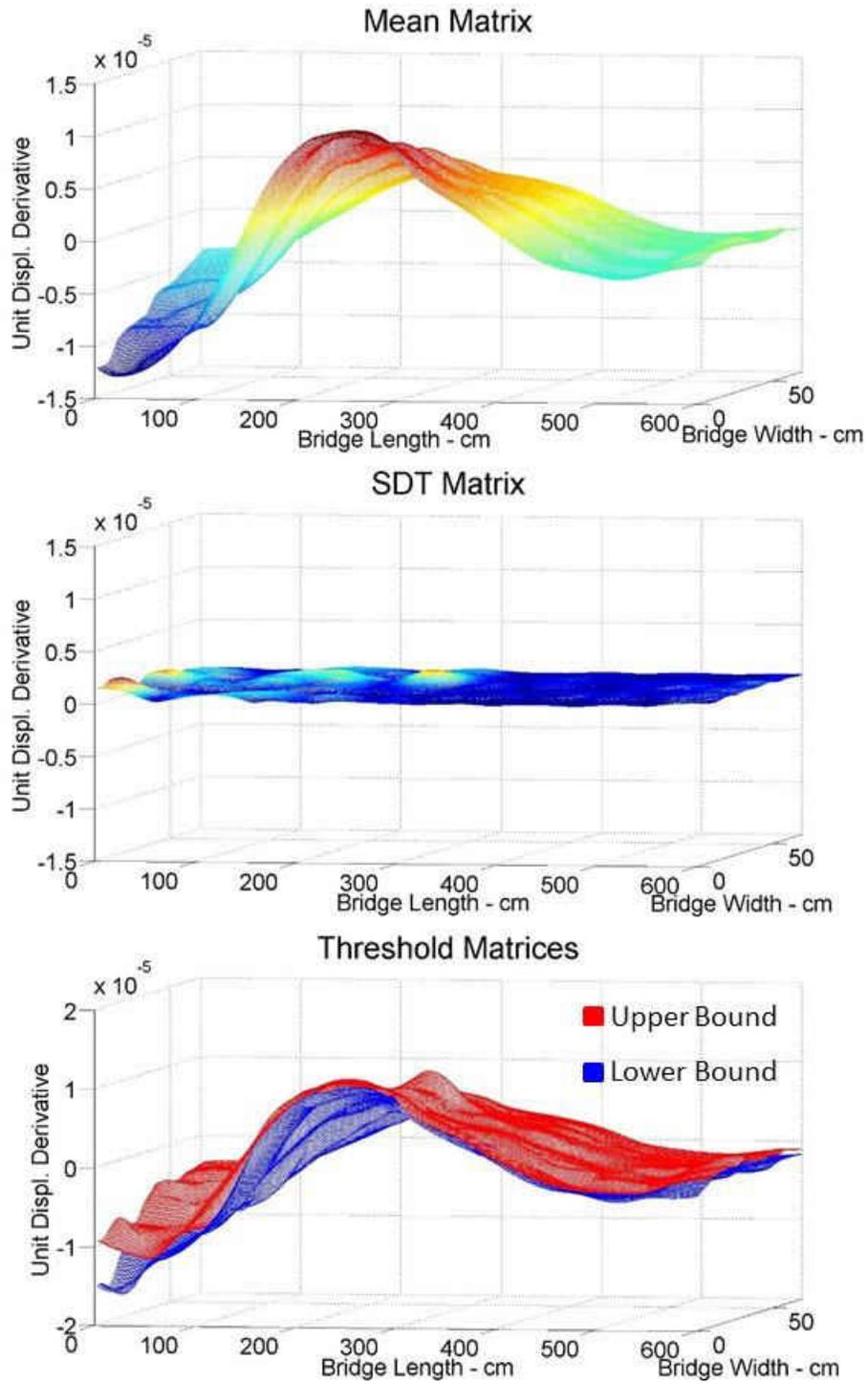


Figure 73: The mean, standard deviation and threshold matrices for the *Baseline* status

Damage Detection

A given UIS-Y matrix is compared with the developed *upper-bound* and *lower-bound* matrices for classifying whether the input data belongs to damaged or healthy state. An example comparison is shown in Figure 74. It is seen that the input data (magenta color surface) has vast amount of its region being out of the range between the threshold matrices (red and blue surfaces). This observation makes the input data classified as belonging to a damage state. For clearer observation, the section cut of those surfaces is illustrated in Figure 75. In that graph, the mean, upper-bound, lower-bound, and input data (latter ranked as damage data) lines are all presented. The graph also shows that whenever the input line (magenta line) lays outside the range of bounds (red and blue lines), the classifier gives alarm for damage that is denoted by dashed black line.

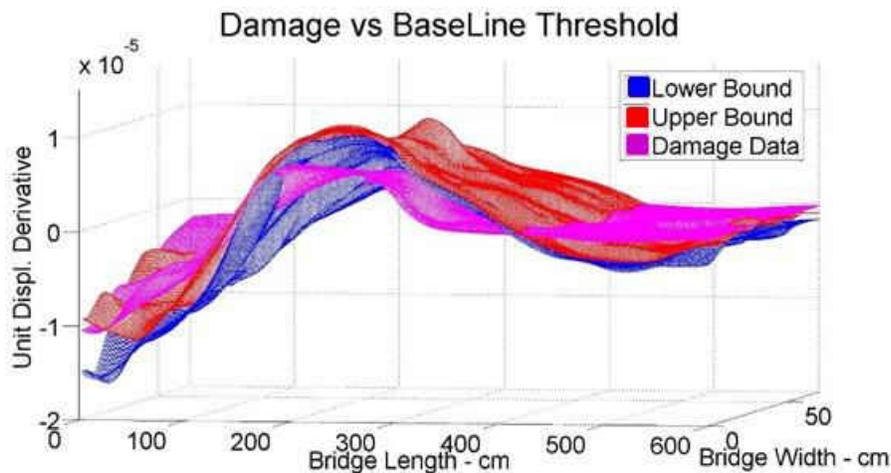


Figure 74: Damage recognition for an input data based on pre-developed thresholds of healthy condition

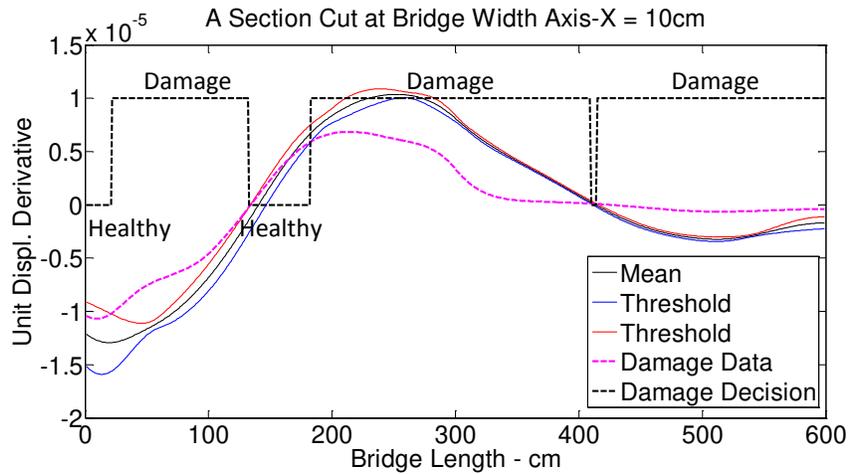


Figure 75: A section cut of damage identification process

The proposed damage identification is verified for four (4) simulated damage scenarios including *Damage 1*, *2*, *3*, and *4*. Following this method, only the data sets from limited four (4) out of twelve (12) measurement locations *L2*, *L5*, *L8*, and *L11* that are corresponding to the mid-span positions of the girders are deployed requiring no sensor network. For each damage scenario, for instance *Damage 1*, the mean matrix of the obtained UIS-Y matrices is calculated for each measurement location. Each particular mean matrix is processed as an input data matrix following the process mentioned previously to rank every *Cell* as either damaged or non-damaged. The results of that classification for the *Damage 1* scenario are illustrated in Figure 76 for the locations *L2*, *L5*, *L8*, and *L11*, respectively. It is seen that 68% of *Cells* of the mean UIS-Y matrix obtained at *L2* is categorized as damaged. The outcomes for the mean UIS-Y matrices acquired from *L5*, *L8*, and *L11* are 65%, 65% and 53%, respectively. The same trend is repeated for the remaining damage scenarios; and the results are illustrated in Figure 77, Figure 78, and Figure 79 corresponding to *Damage 2*, *Damage 3*, and *Damage 4*. The percentages of damaged *Cells* for all damage cases are also presented in Table 13.

Table 13: Percentage of damage *Cells* for damage scenario

	L2 (%)	L5 (%)	L8 (%)	L11 (%)	<i>Average (%)</i>
Damage 1	68	65	55	53	60
Damage 2	84	62	86	81	78
Damage 3	23	19	25	18	21
Damage 4	24	19	34	26	26

Table 13 shows that an average of 60% of Cells in the UIS-Y matrix points out damage for the case of *Damage 1*. The percentage is even higher for the case of *Damage 2* that is up to 78% damage alarm. Those results are plausible since both the *Damage 1* and *Damage 2* are global damages that commonly involve critical deterioration. Obviously, the severity of the *Damage 2* is higher than the one obtained from the *Damage 1* due to more deterioration problems reproduced in the *Damage 2*. The local damages such as *Damage 3* and *Damage 4* show much lower damage percentages, for example, 21% and 26% respectively. Again, the damage percentage value of the *Damage 4* scenario (26%) is higher than the one determined from the *Damage 3* data (21%) since the two local damage positions are simulated together in the *Damage 4* whereas only one deterioration position exist in the *Damage 3*.

The proposed framework for damage detection based on the UIS-Y *Cells* classification is verified. The method can clearly recognize both types of damage: global and local by quantifying the classified damage area on the bridge deck coordinate system, for example, from 21% to 78% in this study. In the next step, the framework is extended for damage localization by quantifying the change between the input UIS-Y data and the healthy threshold matrices for each *Cells*. The results of this implementation are described in the subsequent section.



Figure 76: Damage area for Damage 1

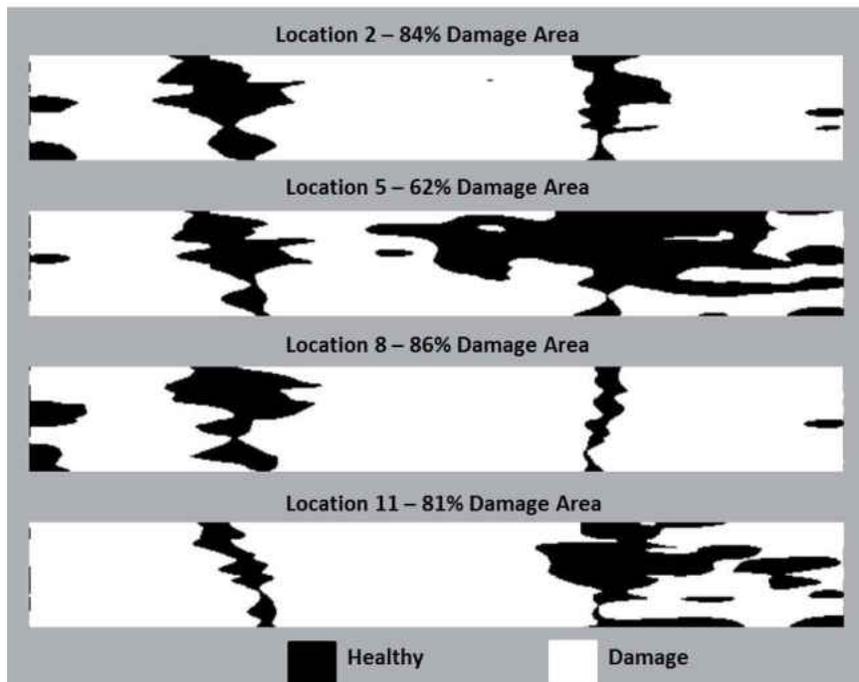


Figure 77: Damage area for Damage 2

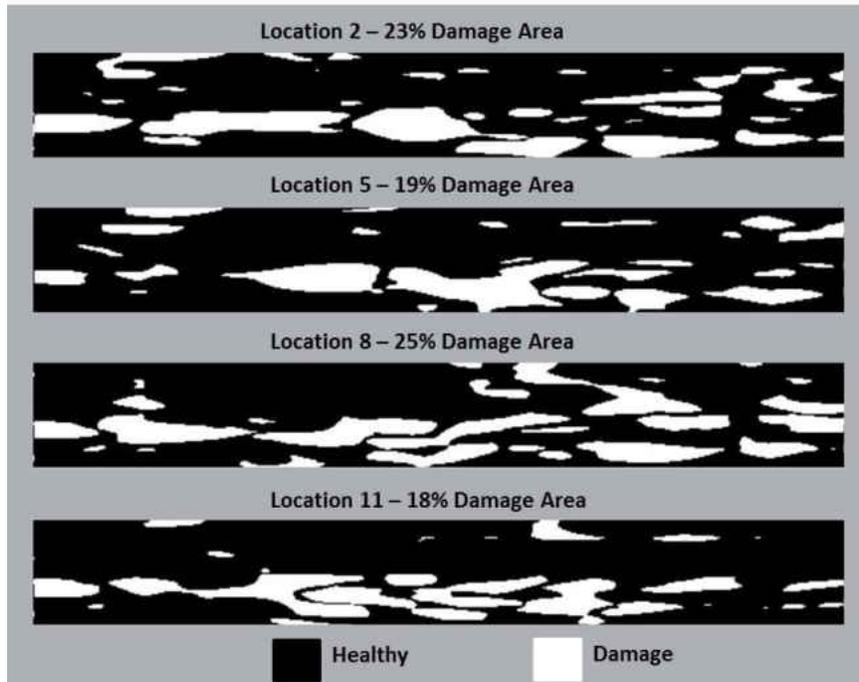


Figure 78: Damage area for Damage 3



Figure 79: Damage area for Damage 4

Damage Localization

The alteration between two conditions of the structure (e.g. healthy status vs. damage status) is determined based on the subtraction of the *Baseline* mean matrix M^B from the *Damage* mean matrix M^D , which is denoted as follows:

$$\Delta = |M^B - M^D| \quad (72)$$

Where Δ is the alteration matrix between two structural conditions; M^B is the mean of the series of the UIS-Y matrices obtained in the *Baseline* condition; M^D is the mean of the series of the UIS-Y matrices acquired in the *Damage* condition

Once damage is detected in terms of the scheme described in the previous context, it is located by scanning the matrix Δ to find out the region with high intensity values. To address this objective, another type of threshold that equals to 90% of the maximum value in the matrix Δ is introduced. The proposed framework is once more verified for four (4) reproduced damage scenarios: *Damage 1*, *2*, *3*, and *4*. The identification results, which are obtained by analyzing the data from only four (4) measurement locations $L2$, $L5$, $L8$, and $L11$ of damage regions for the *Damage 1* are illustrated in Figure 80. Although only limited (4) measurement locations are deployed, the damage location is successfully detected by the measurement data from the $L2$ and $L5$. The observation is reasonable since the damage is reproduced on the Girder 1 where the $L2$ and $L5$ be. The *Damage 2* is also verified and the outcomes are depicted in Figure 81. It is seen that the positions of damage can be pinpointed by the data obtained from locations $L2$, $L8$, and $L11$. A limitation to be remarked from these results is that several false positives are detected as damage locations. Although some false positives can be discarded as being “too small damage area” or the “non-structural damage area”, the other false positives need further processing for rejection.

The method is also conducted for the local damages including the *Damage 3* and *Damage 4*. Figure 82 shows the damage localization results for the *Damage 3* scenario; and unfortunately, the proposed approach is not really successful for this case. Although the damage location seem to be detected from the data at the location *L2*, numerous false positives are also identified. However, the results are better for the *Damage 4* condition. Figure 83 reveals that the damage positions are detected by processing the data from locations *L8* and *L11*; and it is plausible since the damage is simulated on the Girder 2. However, it is also seen that numerous false positives are obtained for this damage scenario as well.

The proposed framework for identifying damage by deploying limited measurement locations (or sensors) is verified in this section. With only four (4) measurement locations, the first derivative of the UIS is exploited for not only damage detection but also damage localization. The analyzed outcomes show that the new approach, which eventually outperforms the conventional way based on an array of sensors, can efficiently detect the global damage as well as the local damage,. For the aim of damage localization, the proposed method can clearly locate the global damage. However, it meets challenges with local damage cases since numerous false positives of damage locations are also recognized for those cases.



Figure 80: Damage localization for Damage 1



Figure 81: Damage localization for Damage 2



Figure 82: Damage localization for Damage 3



Figure 83: Damage localization for Damage 4

Conclusions

The main goal of this chapter is to propose two different frameworks for damage identification for bridge structures using the new damage indicator, UIS. The content herein comprises two parts corresponding to those damage identification methods. In the first method, the classical approach is followed since damage in structures can be detected and then be localized based on a network of sensors (or measurement locations). The main aim of the first implementation is to verify the consistency and reliability of the new damage indicator UIS. The second method is a novel approach that takes advantage of the geometric components of the UIS. Thus, the second framework can do both detection and localization without the need for a measurement location array. Since the new proposed framework does not require a network of sensors, it enables the possibility of a low-cost and convenient SHM implementation.

Both damage identification frameworks are verified in the UCF 4–Span bridge, a total of four (4) damage scenarios are simulated including two (2) global damage cases (*Damage 1 & Damage 2*) and other two (2) local damage cases (*Damage 3 & Damage 4*). For the conventional framework based on the twelve (12) measurement locations, the damage can be clearly detected and localized for the global damage case. However, for the local damage scenarios, the results are not so clear for damage localization. Especially for the *Damage 3*, the fuzzy clues and proofs can be seen for both implementations: damage detection and damage localization.

The new damage identification framework presents a great performance for damage detection even though only the data acquired from four (4) measurement locations is utilized. The results obtained by the new approach for local damage cases are even better than the ones processed via the classical approach. The damage localization method yields good results for the global damage scenarios. Unfortunately, that implementation produces numerous false positives

of damage locations for the local damage cases. Thus, a more robust algorithm for damage localization based on this approach should be studied for rejecting the false positives accurately.

CHAPTER SIX: CONCLUSIONS

Summary

This research proposes a novel Structural Identification (St-Id) approach for bridges based on camera technologies and computer vision methodologies. The collected data employed in the St-Id system are acquired by series of computer vision based measurements. Subsequently, the input and output data sets are analyzed to construct a new types of damage indicator, named *Unit Influence Surface (UIS)*. Finally, the new damage detection and localization framework is introduced that does not require a network of sensors, but much less number of sensors. The research has four main sections presented from *Chapter 2* to *5*, respectively. Brief review of each *Chapter* is noted as follows.

Fully non-contact displacement measurement by means of image key-points:

A measurement method is proposed to obtain displacements and vibrations of structures using a low-cost camera and computer vision techniques. The advantage of this proposed method is that it does not require any type of physical targets attached on structures, which are commonly required by current vision-based methods. This objective is addressed by employing the image key-points as a new type of virtual markers instead of physical targets. The key-points of measurement locations are extracted by using different robust computer vision algorithms, and their characteristics show a potential ability to replace physical targets. Key-point matches among image frames are further improved by using an outlier detection algorithm to discard the false matches. In addition, a specific camera calibration procedure is developed to calculate the unit conversion ratio R between the image coordinates (pixels) and the world coordinates (mm) since conventional targets no longer exist.

The proposed computer vision-based method is developed and verified on laboratory bridge model, a real life bridge, a stadium under different experimental conditions including altering light conditions, different camera locations (distances and angles), and camera frame rates. Since this method does not require any type of attachment, its implementation is observed to be much more convenient (e.g. faster and easier) than other conventional sensors employed on the same structure. For verification purposes, the displacement data sets obtained by the proposed vision-based method are compared to conventional LVDT data sets. The comparison results show very high correlation and similarity by means of statistical measures such as correlation coefficient ρ (from 0.929 to 0.989) and the determination of *R-squared* coefficient (from 0.9368 to 0.9998) for both laboratory and real-life implementations. In addition, natural frequencies of modes (or modal frequencies) are also identified from the dynamic displacement time histories as the cameras utilized in this study can capture images at speed/frame rates of 30 Hz and 60 Hz. Identified natural frequencies are compared to the frequencies obtained using accelerometer data. The comparison of these two aforementioned frequency sets shows a maximum difference of 2.44 % indicating that the proposed method can successfully determine deflections and natural frequencies.

Accuracy of the proposed vision-based method is evaluated in the laboratory. The evaluation reveals that the measurement accuracy highly depends on the distance between the camera and monitoring positions. For example, the method accuracy is ± 0.01 mm at a distance of 3m; but the accuracy decreases to ± 0.04 mm for a measurement distance of 13 m. The relationship between the accuracy of the proposed monitoring method and the camera distance is demonstrated a linear function.

Estimation of traffic vehicle weights and locations based on computer vision algorithms:

A framework for estimating the loading (amplitudes and locations) induced by traffic vehicles passing over a bridge is introduced by employing computer vision techniques. The highlight of the proposed framework is that neither bridge closure nor any conventional sensors are needed. A surveillance camera is placed at a relatively higher position on the bridge for acquiring images of traffic vehicles moving on the deck. Those images are analyzed to identify the vehicles and then the images are categorized into classes based on a series of advanced computer vision algorithms and features including AdaBoost technique, cascade classifier and HOG descriptors. Since the vehicle classes are pre-defined, the weight distribution of each class is available, thus this weight is assigned to the categorized vehicle. The detected vehicles are also labeled by bounding boxes corresponding to their types. Eventually, the positions of these bounding boxes (pixel metric) in image coordinates are converted to the positions defined by real-life coordinate system affixed on the bridge deck for localizing the vehicle positions. The transformation process is achieved by adjusting the camera model equation based on the assumption that the vehicles move on a planar bridge deck.

The proposed framework is validated through tests conducted in the UCF Structural Lab on an experimental bridge. Multiple small-scale vehicles are deployed to simulate traffic on the bridge. Those vehicles are identified and categorized into their classes (*Class 1*, *Class 2*, or *Class 3*) at the detection rates ranging from 87.2 % to 95.9 %. Subsequently, the detected vehicles are localized by using the transformation matrix T , which is determined by means of employing 15 reference points on the bridge deck. The transformation matrix is verified for re-calculation of the reference point locations in the world coordinate system from their places on images. The accuracy of reference point coordinates yields a maximum relative error of 1.2 %.

A new damage indicator for bridge St-Id system: Unit Influence Surface:

A new structural identification (St-Id) approach is formulated with both input (vehicle weights and locations) and output (structural displacements) data being acquired by a series of computer vision-based measurements. A new damage indicator involving structural parameters for Bridge St-Id system is introduced. The proposed damage indicator is named Unit Influence Surface (UIS), which is a 3-D version of Unit Influence Line (UIL). A UIS is constructed by deploying surface fitting algorithms (e.g. Bilinear Interpolation and Thin-plate Spline Interpolation) to a set of *UIS-points*, each of which describes a state of *vehicle loading-structural response* (input-output) interaction while traffic passes on the bridge deck. A UIS-point (X_j, Y_j, U_j) is developed from a *configuration location* (X_j, Y_j) derived from vehicle positions and a *normalized displacement* U_j at the measurement location. While *the configuration location* can be calculated as a centroid of the detected vehicle positions, *the normalized displacement* is determined by operating an iterative numerical procedure using the coupled information of the vehicle and structural displacement data.

The UIS's are extracted for the 4-Span Bridge in UCF Structural Lab. To validate the outcomes of the proposed protocol, a total of twelve (12) positions under main girder flanges are selected for constructing the UIS's. Since these positions are placed symmetrically over two symmetrical axes (middle-support axis and median of the deck), the UIS's at the symmetrical locations illustrate identical shapes and maxima values (error values from 0.2% to 7.2%). The UIS consistency is also verified since it is the most essential characteristics for a damage indicator. A consistent verification method is deployed by constructing numerous (15) UIS's for each measurement locations from a database of forty (40) tests. A measure of data dispersion for the UIS parameters (e.g. UIS volume and UIS maximum data) yields the maximum of CV (coefficient

of variation) value to be 1.70% and many other CV values to be less than 1.00%. Quantitative analysis of the obtained UIS confirms the consistency and reliability. With these characteristics, the UIS promises a great potential as a sensitive damage indicator.

Bridge damage detection and localization based on limited measurement locations (sensors):

The last section of this dissertation is to propose two different frameworks for damage identification for bridge structures using the new damage indicator, UIS. In the first method, the classical approach is followed since damage in structures can be detected and then be localized based on a network of sensors (or measurement locations). The main aim of the first implementation is to verify the consistency and reliability of the new damage indicator UIS. The second method is a novel approach that takes advantage of the geometric components of the UIS. The damage detection is deployed at every point (X_j, Y_j) of the UIS on the bridge coordinate system to finally decide the bridge either damage or non-damage. Thus, the second framework can do both detection and localization without the need for a measurement location array.

Both damage identification frameworks are verified in the UCF 4–Span bridge, a total of four (4) damage scenarios are simulated including two (2) global damage cases (*Damage 1 & Damage 2*) and other two (2) local damage cases (*Damage 3 & Damage 4*). For the conventional approach based on the twelve (12) measurement locations, the damage can be clearly detected and localized for the global damage case. However, for the local damage scenarios, the results are not so clear for damage localization. Especially for the *Damage 3*, the fuzzy clues and proofs can be seen for both implementations: damage detection and damage localization.

The new damage identification framework presents a great performance for damage detection even though only the data acquired from four (4) measurement locations are utilized.

The results obtained by the new approach for local damage cases are even better than the ones processed via the classical approach. The damage localization method yields good results for the global damage scenarios. However, that implementation produces numerous false positives of damage locations for the local damage cases.

Contributions

Several main contributions are observed in this dissertation. Those are listed follows.

- ✓ Vehicle Load (Input) Modeling Using Computer Vision: To the best knowledge of the literature search, it is the first time that the vehicles will be detected, tracked and classified including wheel weights, axle spacing, and moving load locations, by combining a series of computer vision techniques. In other words, the novelty of the study is developing algorithms for obtaining equivalent moving loads on the bridge from complex loading pattern due to multiple vehicles using the camera data.
- ✓ Bridge Response (Output) Using Full Non-contact Approach Using Video/Image Processing: Besides using conventional sensors to extract bridge response, a full non-contact method for displacement measurement is explored by utilizing computer vision techniques. The method promises further simplicity in deployment and cost-effectiveness as the response will be measured without any targets attached to the structure.
- ✓ Image-based Structural Identification (St-Id) using Input-Output Measurements: The video data processed using computer vision methods to determine the vehicle loads are analyzed in the context of structural identification (St-Id), which has not been available in the literature

- ✓ Development and Demonstration of a New Damage Indicator: A new damage indicator Unit Influence Surface (UIS) for bridge St-Id systems is introduced using the data from a series non-contact measurements. Thus, a bridge does not required to be closed while a monitoring study is being conducted. Furthermore, non-contact measurements are cost-effective, safe, and time-saving implementation.
- ✓ Broader Impact on the Structural Assessment of Bridges: Based on the outcome of this research, a conventional network of sensors may no longer be necessity for certain conditions. This research enables reduction of sensors, installation time and cost; and as a result, the bridge monitoring practices may become simpler, faster, and cost-effective for many bridges.

Recommendations for Future Research

Some challenges are pointed out, especially for real-life implementation issues that need to be considered and addressed are as follows.

- ✓ Extracting and matching key-points suffer from intensive computation. Thus, the vision-based displacement measurements cannot be a real-time implementation. The image data is acquired in fields, and latter tediously processed, especially for long-term monitoring data.
- ✓ The vision-based displacement measurement method can only monitor a single position in real-life structures. A multiple measurement point system is a challenge involving synchronization issue among cameras.

- ✓ Developing a comprehensive database for classes of real-life vehicles is a cumbersome job. The traffic vehicle information and their classes require an extensive research and data collection.
- ✓ Even though a comprehensive database of traffic vehicles is developed, the weight of a classified vehicle may be very different from its class weight. This challenge may be mitigated by long-term monitoring or by using a portable WIM (Weigh In Motion) system, which is recently developed (Kwon, 2012).
- ✓ Calculation of the UIS is a complicated process. It may be a trade-off between advantages of field data acquisition and complicated analysis.

LIST OF REFERENCES

- Aktan, E., Chase, S., Inman, D., & Pines, D. (2001, March 6-8). *Monitoring and Managing the Health of Infrastructure Systems*. Paper presented at the SPIE Conference on Health Monitoring of Highway Transportation Infrastructure, Irvine, CA.
- Alahi, A., Ortiz, R., & Vanderghenst, P. (2012). *Freak: Fast retina keypoint*. Paper presented at the Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on.
- Arangio, S., & Beck, J. (2012). Bayesian neural networks for bridge integrity assessment. *Structural Control and Health Monitoring*, 19(1), 3-21.
- Arróspide, J., Salgado, L., Nieto, M., & Jaureguizar, F. (2008). *On-board robust vehicle detection and tracking using adaptive quality evaluation*. Paper presented at the Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on.
- Baehring, D., Simon, S., Niehsen, W., & Stiller, C. (2005). *Detection of close cut-in and overtaking vehicles for driver assistance based on planar parallax*. Paper presented at the Intelligent Vehicles Symposium, 2005. Proceedings. IEEE.
- Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding*, 110(3), 346-359. doi: 10.1016/j.cviu.2007.09.014
- Brincker, R., Zhang, L., & Andersen, P. (2001). Modal identification of output-only systems using frequency domain decomposition. *Smart Materials and Structures*, 10(3), 441.
- Brox, T., Bruhn, A., Papenber, N., & Weickert, J. (2004). High accuracy optical flow estimation based on a theory for warping *Computer Vision-ECCV 2004* (pp. 25-36): Springer.
- Buch, N., Velastin, S., & Orwell, J. (2011). A review of computer vision techniques for the analysis of urban traffic. *Intelligent Transportation Systems, IEEE Transactions on*, 12(3), 920-939.
- Busca, G., Cigada, A., Mazzoleni, P., Tarabini, M., & Zappa, E. (2013a). *Static and Dynamic Monitoring of Bridges by means of Vision-based Measuring System*. Paper presented at the IMAC XXXII A Conference and Exposition on Structural Dynamics.
- Busca, G., Cigada, A., Mazzoleni, P., Tarabini, M., & Zappa, E. (2013b). Static and Dynamic Monitoring of Bridges by Means of Vision-Based Measuring System *Topics in Dynamics of Bridges, Volume 3* (pp. 83-92): Springer.
- Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (2010). Brief: Binary robust independent elementary features *Computer Vision-ECCV 2010* (pp. 778-792): Springer.

- Cantatore, A., Cigada, A., Sala, R., & Zappa, E. (2009). Hyperbolic tangent algorithm for periodic effect cancellation in sub-pixel resolution edge displacement measurement. *Measurement*, 42(8), 1226-1232. doi: 10.1016/j.measurement.2009.06.001
- Caprioli, A., Reynolds, P., & Vanali, M. (2007). Evaluation of serviceability assessment measures for different stadia structures and different live concert events. *IMAC XXV. Orlando, Florida, USA*.
- Catbas, F., Kijewski-Correa, T., & Aktan, A. (2011). *Structural identification (St-Id) of constructed facilities: Approaches, methods and technologies for effective practice of St-Id*. Paper presented at the Am Soc Civ Eng.
- Catbas, F., Lenett, M., Brown, D., Doebling, S., Farrar, C., & Turer, A. (1997). *Modal analysis of multi-reference impact test data for steel stringer bridges*. Paper presented at the PROCEEDINGS-SPIE THE INTERNATIONAL SOCIETY FOR OPTICAL ENGINEERING.
- Catbas, F. N., & Aktan, A. E. (2002). Condition and damage assessment: issues and some promising indices. *Journal of Structural Engineering*.
- Catbas, F. N., Brown, D. L., & Aktan, A. E. (2004). Parameter estimation for multiple-input multiple-output modal analysis of large structures. *Journal of Engineering Mechanics*.
- Catbas, F. N., Brown, D. L., & Aktan, A. E. (2006). Use of modal flexibility for damage detection and condition assessment: case studies and demonstrations on large structures. *Journal of Structural Engineering*.
- Catbas, F. N., Gokce, H. B., & Gul, M. (2012). Nonparametric analysis of structural health monitoring data for identification and localization of changes: Concept, lab, and real-life studies. *Structural Health Monitoring*, 11(5), 613-626. doi: 10.1177/1475921712451955
- Catbas, F. N., Gul, M., & Burkett, J. L. (2008). Conceptual damage-sensitive features for structural health monitoring: Laboratory and field demonstrations. *Mechanical Systems and Signal Processing*, 22(7), 1650-1669. doi: 10.1016/j.ymssp.2008.03.005
- Catbas, F. N., Gul, M., & Burkett, J. L. (2008). Damage assessment using flexibility and flexibility-based curvature for structural health monitoring. *Smart Materials and Structures*, 17(1), 015024. doi: 10.1088/0964-1726/17/01/015024
- Catbas, F. N., Gul, M., & Sazak, H. (2009). *Dynamic Response Monitoring and Correlation to Crowd Movement at a Football Stadium*. Paper presented at the 27th International Modal Analysis Conference (IMAC XXVII).
- Catbas, F. N., Gul, M., Zaurin, R., Gokce, H. B., Terrell, T., Dumlupinar, T., & Maier, D. (2010). Long term bridge maintenance monitoring demonstration on a movable bridge: A framework for structural health monitoring of movable bridges.

- Catbas, F. N., & Kijewski-Correa, T. (2013). Structural identification of constructed systems: Collective effort toward an integrated approach that reduces barriers to adoption. *Journal of Structural Engineering*, 139(10), 1648-1652.
- Catbas, F. N., Zaurin, R., Gul, M., & 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. doi: 10.1061/(asce)be.1943-5592.0000288
- Cavadas, F., Smith, I. F., & Figueiras, J. (2013). Damage detection using data-driven methods applied to moving-load responses. *Mechanical Systems and Signal Processing*, 39(1), 409-425.
- Cedras, C., & Shah, M. (1995). Motion-based recognition a survey. *Image and Vision Computing*, 13(2), 129-155.
- Chen, Z.-W., Zhu, S., Xu, Y.-L., Li, Q., & Cai, Q.-L. (2014). Damage detection in long suspension bridges using stress influence lines. *Journal of Bridge Engineering*, 20(3), 05014013.
- Chen, Z., & Ellis, T. (2011). *Multi-shape descriptor vehicle classification for urban traffic*. Paper presented at the Digital Image Computing Techniques and Applications (DICTA), 2011 International Conference on.
- Chen, Z., Ellis, T., & Velastin, S. (2012). *Vehicle detection, tracking and classification in urban traffic*. Paper presented at the Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on.
- Cheon, M., Lee, W., Yoon, C., & Park, M. (2012). Vision-based vehicle detection system with consideration of the detecting location. *Intelligent Transportation Systems, IEEE Transactions on*, 13(3), 1243-1252.
- Dalal, N., & Triggs, B. (2005). *Histograms of oriented gradients for human detection*. Paper presented at the Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on.
- Dessauer, M. P., & Dua, S. (2010). *Optical flow object detection, motion estimation, and tracking on moving vehicles using wavelet decompositions*. Paper presented at the SPIE Defense, Security, and Sensing.
- Doebling, S., Farrar, C., Prime, M., & Shevitz, D. (1996). Damage identification in structures and mechanical systems based on changes in their vibration characteristics: a detailed literature survey. *Los Alamos National Laboratory Rep. No. LA-13070-MS*.
- Duchon, J. (1977). Splines minimizing rotation-invariant semi-norms in Sobolev spaces *Constructive theory of functions of several variables* (pp. 85-100): Springer.

- Elgamal, A., Conte, J., Masri, S., Fraser, M., Fountain, T., Gupta, A., . . . El Zarki, M. (2003). *Health monitoring framework for bridges and civil infrastructure*. Paper presented at the Proceedings of the 4th International Workshop on Structural Health Monitoring.
- Farrar, C. R., & Worden, K. (2012). *Structural health monitoring: a machine learning perspective*: John Wiley & Sons.
- Felzenszwalb, P. F., Girshick, R. B., McAllester, D., & Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9), 1627-1645.
- Fraser, M. S. (2006). *Development and implementation of an integrated framework for structural health monitoring*. (Doctoral Dissertation), University of California at San Diego.
- Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1), 119-139.
- Głowacz, A., Mikrut, Z., & Pawlik, P. (2012). Video detection algorithm using an optical flow calculation method *Multimedia Communications, Services and Security* (pp. 118-129): Springer.
- González, A., Rowley, C., & O'Brien, E. J. (2008). A general solution to the identification of moving vehicle forces on a bridge. *International journal for numerical methods in engineering*, 75(3), 335-354.
- Gul, M., & Catbas, F. N. (2008). Ambient vibration data analysis for structural identification and global condition assessment. *Journal of Engineering Mechanics*, 134(8), 650-662.
- Gul, M., & Catbas, F. N. (2009). Statistical pattern recognition for Structural Health Monitoring using time series modeling: Theory and experimental verifications. *Mechanical Systems and Signal Processing*, 23(7), 2192-2204.
- Gul, M., & Catbas, F. N. (2011). Structural health monitoring and damage assessment using a novel time series analysis methodology with sensor clustering. *Journal of Sound and Vibration*, 330(6), 1196-1210. doi: 10.1016/j.jsv.2010.09.024
- Gul, M., Catbas, F. N., & Hattori, H. (2013). Image-Based Monitoring of Open Gears of Movable Bridges for Condition Assessment and Maintenance Decision Making. *Journal of Computing in Civil Engineering*, 04014034. doi: 10.1061/(asce)cp.1943-5487.0000307
- Gupte, S., Masoud, O., Martin, R. F., & Papanikolopoulos, N. P. (2002). Detection and classification of vehicles. *Intelligent Transportation Systems, IEEE Transactions on*, 3(1), 37-47.
- Harris, C., & Stephens, M. (1988). *A combined corner and edge detector*. Paper presented at the Alvey vision conference.

- Haselhoff, A., Schauland, S., & Kummert, A. (2008). *A signal theoretic approach to measure the influence of image resolution for appearance-based vehicle detection*. Paper presented at the Intelligent Vehicles Symposium, 2008 IEEE.
- Horn, B. K., & Schunck, B. G. (1981). *Determining optical flow*. Paper presented at the 1981 Technical symposium east.
- Hsieh, J.-W., Yu, S.-H., Chen, Y.-S., & Hu, W.-F. (2006). Automatic traffic surveillance system for vehicle tracking and classification. *Intelligent Transportation Systems, IEEE Transactions on*, 7(2), 175-187.
- Hu, C., & Afzal, M. T. (2006). A statistical algorithm for comparing mode shapes of vibration testing before and after damage in timbers. *Journal of Wood Science*, 52(4), 348-352.
- Huang, C.-L., & Liao, W.-C. (2004). *A vision-based vehicle identification system*. Paper presented at the null.
- Im, S. B., Hurlbaeus, S., & Kang, Y. J. (2011). Summary review of GPS technology for structural health monitoring. *Journal of Structural Engineering*, 139(10), 1653-1664.
- Jahanshahi, M. R., & Masri, S. F. (2012). Adaptive vision-based crack detection using 3D scene reconstruction for condition assessment of structures. *Automation in Construction*, 22, 567-576.
- Jauregui, D., White, K., Woodward, C., & Leitch, K. (2003). Noncontact Photogrammetric Measurement of Vertical Bridge Deflection. *Journal of Bridge Engineering*. doi: 10.1061//asce/1084-0702/2003/8:4/212
- Javed, O., Rasheed, Z., Shafique, K., & Shah, M. (2003). *Tracking across multiple cameras with disjoint views*. Paper presented at the Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on.
- Javed, O., & Shah, M. (2002). Tracking and object classification for automated surveillance *Computer Vision—ECCV 2002* (pp. 343-357): Springer.
- Jazayeri, A., Cai, H., Zheng, J. Y., & Tuceryan, M. (2011). Vehicle detection and tracking in car video based on motion model. *Intelligent Transportation Systems, IEEE Transactions on*, 12(2), 583-595.
- Jurjo, D. L. B. R., Magluta, C., Roitman, N., & Gonçalves, P. B. (2010). Experimental methodology for the dynamic analysis of slender structures based on digital image processing techniques. *Mechanical Systems and Signal Processing*, 24(5), 1369-1382. doi: 10.1016/j.ymsp.2009.12.006
- Kabir, S., Rivard, P., He, D.-C., & Thivierge, P. (2009). Damage assessment for concrete structure using image processing techniques on acoustic borehole imagery. *Construction and Building Materials*, 23(10), 3166-3174.

- Kanhere, N. K., & Birchfield, S. T. (2008). Real-time incremental segmentation and tracking of vehicles at low camera angles using stable features. *Intelligent Transportation Systems, IEEE Transactions on*, 9(1), 148-160.
- Kao, C. Y., & Loh, C. H. (2013). Monitoring of long-term static deformation data of Fei-Tsui arch dam using artificial neural network-based approaches. *Structural Control and Health Monitoring*, 20(3), 282-303.
- Khan, S. M., Cheng, H., Matthies, D., & Sawhney, H. (2010). *3D model based vehicle classification in aerial imagery*. Paper presented at the Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.
- Kim, G., & Cho, J.-S. (2012). *Vision-based vehicle detection and inter-vehicle distance estimation*. Paper presented at the Control, Automation and Systems (ICCAS), 2012 12th International Conference on.
- Kim, S.-W., & Kim, N.-S. (2013). Dynamic characteristics of suspension bridge hanger cables using digital image processing. *NDT & E International*, 59, 25-33. doi: 10.1016/j.ndteint.2013.05.002
- Kim, S.-W., Lee, S.-S., Kim, N.-S., & Kim, D.-J. (2013). Numerical model validation for a prestressed concrete girder bridge by using image signals. *KSCE Journal of Civil Engineering*, 17(3), 509-517. doi: 10.1007/s12205-013-0560-1
- Kim, S. W., Jeon, B. G., Kim, N. S., & Park, J. C. (2013). Vision-based monitoring system for evaluating cable tensile forces on a cable-stayed bridge. *Structural Health Monitoring*, 12(5-6), 440-456. doi: 10.1177/1475921713500513
- Kim, Z. (2008). *Real time object tracking based on dynamic feature grouping with background subtraction*. Paper presented at the Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on.
- Kwon, T. M. (2012). Development of a Weigh-Pad-Based Portable Weigh-In-Motion System. Minnesota Department of Transportation.
- Laory, I., Trinh, T. N., & Smith, I. F. (2011). Evaluating two model-free data interpretation methods for measurements that are influenced by temperature. *Advanced Engineering Informatics*, 25(3), 495-506.
- Lee, J. J., Fukuda, Y., Shinozuka, M., Yun, C.-B., & Cho, S. (2007). Development and application of a vision-based displacement measurement system for structural health monitoring of civil structures. *Smart Structures and Systems*.
- Lee, J. J., Lee, J. W., Yi, J. H., Yun, C. B., & Jung, H. Y. (2005). Neural networks-based damage detection for bridges considering errors in baseline finite element models. *Journal of Sound and Vibration*, 280(3), 555-578.

- Lee, J. J., & Shinozuka, M. (2006). A vision-based system for remote sensing of bridge displacement. *NDT & E International*, 39(5), 425-431. doi: 10.1016/j.ndteint.2005.12.003
- Leutenegger, S., Chli, M., & Siegwart, R. Y. (2011). *BRISK: Binary robust invariant scalable keypoints*. Paper presented at the Computer Vision (ICCV), 2011 IEEE International Conference on.
- Levinson, J., Askeland, J., Dolson, J., & Thrun, S. (2011). *Traffic light mapping, localization, and state detection for autonomous vehicles*. Paper presented at the Robotics and Automation (ICRA), 2011 IEEE International Conference on.
- Llorca, D., Arroyo, R., & Sotelo, M. (2013). *Vehicle logo recognition in traffic images using hog features and svm*. Paper presented at the Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on.
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91-110.
- Lu, Y., & Gao, F. (2005). A novel time-domain auto-regressive model for structural damage diagnosis. *Journal of Sound and Vibration*, 283(3), 1031-1049.
- Lucas, B. D., & Kanade, T. (1981). *An iterative image registration technique with an application to stereo vision*. Paper presented at the IJCAI.
- Magalhães, F., Cunha, A., & Caetano, E. (2012). Vibration based structural health monitoring of an arch bridge: from automated OMA to damage detection. *Mechanical Systems and Signal Processing*, 28, 212-228.
- Malekzadeh, M. (2014). *Structural Health Monitoring Using Novel Sensing Technologies and Data Analysis Methods*. University of Central Florida.
- Malekzadeh, M., Atia, G., & Catbas, F. (2015). Performance-based structural health monitoring through an innovative hybrid data interpretation framework. *Journal of Civil Structural Health Monitoring*, 5(3), 287-305.
- Matsumoto, M., Mitani, K., & Catbas, F. N. (2015). *NDE for Bridge Assessment using Image Processing and Infrared Thermography*. Paper presented at the Transportation Research Board 94th Annual Meeting.
- Nair, K. K., Kiremidjian, A. S., & Law, K. H. (2006). Time series-based damage detection and localization algorithm with application to the ASCE benchmark structure. *Journal of Sound and Vibration*, 291(1), 349-368.
- Nandwana, B., & Maiti, S. (1997). Detection of the location and size of a crack in stepped cantilever beams based on measurements of natural frequencies. *Journal of Sound and Vibration*, 203(3), 435-446.

- Ng, L. T., Suandi, S. A., & Teoh, S. S. (2014). *Vehicle Classification Using Visual Background Extractor and Multi-class Support Vector Machines*. Paper presented at the The 8th International Conference on Robotic, Vision, Signal Processing & Power Applications.
- Niknejad, H. T., Takeuchi, A., Mita, S., & McAllester, D. (2012). On-road multivehicle tracking using deformable object model and particle filter with improved likelihood estimation. *Intelligent Transportation Systems, IEEE Transactions on*, 13(2), 748-758.
- O'Brien, E. J., Quilligan, M., & Karoumi, R. (2006). *Calculating an influence line from direct measurements*. Paper presented at the Institution of Civil Engineers.
- O'Brien, E., Carey, C., & Keenahan, J. (2015). Bridge damage detection using ambient traffic and moving force identification. *Structural Control and Health Monitoring*.
- Ohn-Bar, E., Sivaraman, S., & Trivedi, M. (2013). *Partially occluded vehicle recognition and tracking in 3D*. Paper presented at the Intelligent Vehicles Symposium (IV), 2013 IEEE.
- Oliveira, H., & Correia, P. L. (2013). Automatic road crack detection and characterization. *Intelligent Transportation Systems, IEEE Transactions on*, 14(1), 155-168.
- Omenzetter, P., & Brownjohn, J. M. W. (2006). Application of time series analysis for bridge monitoring. *Smart Materials and Structures*, 15(1), 129.
- Orcesi, A. D., & Frangopol, D. M. (2010). Inclusion of crawl tests and long-term health monitoring in bridge serviceability analysis. *Journal of Bridge Engineering*, 15(3), 312-326.
- Park, K.-Y., & Hwang, S.-Y. (2014). An improved Haar-like feature for efficient object detection. *Pattern Recognition Letters*, 42, 148-153.
- Parra, I., Sotelo, M., Llorca, D. F., & Ocaña, M. (2010). Robust visual odometry for vehicle localization in urban environments. *Robotica*, 28(03), 441-452.
- Patsias, S., & Staszewskiy, W. J. (2002). Damage Detection Using Optical Measurements and Wavelets. *Structural Health Monitoring*, 1(1), 5-22. doi: 10.1177/147592170200100102
- Phares, B. M., Washer, G. A., Rolander, D. D., Graybeal, B. A., & Moore, M. (2004). Routine highway bridge inspection condition documentation accuracy and reliability. *Journal of Bridge Engineering*, 9(4), 403-413.
- Ponsa, D., López, A., Lumbreras, F., Serrat, J., & Graf, T. (2005). *3D vehicle sensor based on monocular vision*. Paper presented at the Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE.
- Posenato, D., Kripakaran, P., Inaudi, D., & Smith, I. F. (2010). Methodologies for model-free data interpretation of civil engineering structures. *Computers & Structures*, 88(7), 467-482.

- Posenato, D., Lanata, F., Inaudi, D., & Smith, I. F. (2008). Model-free data interpretation for continuous monitoring of complex structures. *Advanced Engineering Informatics*, 22(1), 135-144.
- Poudel, U. P., Fu, G., & Ye, J. (2005). Structural damage detection using digital video imaging technique and wavelet transformation. *Journal of Sound and Vibration*, 286(4-5), 869-895. doi: 10.1016/j.jsv.2004.10.043
- Rad, R., & Jamzad, M. (2005). Real time classification and tracking of multiple vehicles in highways. *Pattern Recognition Letters*, 26(10), 1597-1607.
- Ren, W.-X., Zhao, T., & Harik, I. E. (2004). Experimental and analytical modal analysis of steel arch bridge. *Journal of Structural Engineering*, 130(7), 1022-1031.
- Rios-Cabrera, R., Tuytelaars, T., & Van Gool, L. (2012). Efficient multi-camera vehicle detection, tracking, and identification in a tunnel surveillance application. *Computer Vision and Image Understanding*, 116(6), 742-753.
- Rodrigues, J., Brincker, R., & Andersen, P. (2004). *Improvement of frequency domain output-only modal identification from the application of the random decrement technique*. Paper presented at the Proc. 23rd Int. Modal Analysis Conference, Dearborn, MI.
- Rucka, M., & Wilde, K. (2005). Crack identification using wavelets on experimental static deflection profiles. *Engineering Structures*, 28(2), 279-288. doi: 10.1016/j.engstruct.2005.07.009
- Salawu, O. (1997). Detection of structural damage through changes in frequency: a review. *Engineering Structures*, 19(9), 718-723.
- Salyards, K. A., & Hanagan, L. M. (2009). Evaluation of Vibration Assessment Criteria and Their Application to Stadium Serviceability. *Journal of performance of constructed facilities*, 24(2), 100-107.
- Sam, K.-T., & Tian, X.-L. (2012). *Vehicle logo recognition using modest adaboost and radial tchebichef moments*. Paper presented at the International Conference on Machine Learning and Computing (ICMLC 2012).
- Santos, C. A., Costa, C. O., & Batista, J. P. (2012). Calibration methodology of a vision system for measuring the displacements of long-deck suspension bridges. *Structural Control and Health Monitoring*, 19(3), 385-404. doi: 10.1002/stc.438
- Senlet, T., & Elgammal, A. (2011). *A framework for global vehicle localization using stereo images and satellite and road maps*. Paper presented at the Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on.

- Shi, J., Xu, X., Wang, J., & Li, G. (2010). Beam damage detection using computer vision technology. *Nondestructive Testing and Evaluation*, 25(3), 189-204. doi: 10.1080/10589750903242525
- Shu, J., Zhang, Z., Gonzalez, I., & Karoumi, R. (2013). The application of a damage detection method using Artificial Neural Network and train-induced vibrations on a simplified railway bridge model. *Engineering Structures*, 52, 408-421.
- Sivaraman, S., & Trivedi, M. M. (2010). A general active-learning framework for on-road vehicle recognition and tracking. *Intelligent Transportation Systems, IEEE Transactions on*, 11(2), 267-276.
- Sivaraman, S., & Trivedi, M. M. (2011). *Combining monocular and stereo-vision for real-time vehicle ranging and tracking on multilane highways*. Paper presented at the Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on.
- Sivaraman, S., & Trivedi, M. M. (2013). Integrated lane and vehicle detection, localization, and tracking: A synergistic approach. *Intelligent Transportation Systems, IEEE Transactions on*, 14(2), 906-917.
- Sivaraman, S., & Trivedi, M. M. (2014). Active learning for on-road vehicle detection: A comparative study. *Machine Vision and Applications*, 25(3), 599-611.
- Sładek, J., Ostrowska, K., Kohut, P., Holak, K., Gaška, A., & Uhl, T. (2013). Development of a vision based deflection measurement system and its accuracy assessment. *Measurement*, 46(3), 1237-1249. doi: 10.1016/j.measurement.2012.10.021
- Sohn, H., Czarnecki, J. A., & Farrar, C. R. (2000). Structural health monitoring using statistical process control. *Journal of Structural Engineering*, 126(11), 1356-1363.
- Sohn, H., & Farrar, C. R. (2001). Damage diagnosis using time series analysis of vibration signals. *Smart Materials and Structures*, 10(3), 446.
- Stöhr, S., Link, M., Rohrman, R., & Rüker, W. (2006). *Damage detection based on static measurements on bridge structures*. Paper presented at the IMAC XXIV Conference and Exposition on Structural Dynamics.
- Sturm, P. F., & Maybank, S. J. (1999). *On plane-based camera calibration: A general algorithm, singularities, applications*. Paper presented at the Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.
- Su, X., Khoshgoftaar, T. M., Zhu, X., & Folleco, A. (2007). Rule-based multiple object tracking for traffic surveillance using collaborative background extraction *Advances in Visual Computing* (pp. 469-478): Springer.

- Sung, Y.-C., Miyasaka, T., Lin, T.-K., Wang, C.-Y., & Wang, C.-Y. (2012). A case study on bridge health monitoring using position-sensitive detector technology. *Structural Control and Health Monitoring*, 19(2), 295-308. doi: 10.1002/stc.436
- Tieu, K., & Viola, P. (2000). *Boosting image retrieval*. Paper presented at the Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on.
- Triggs, B. (1998). Autocalibration from planar scenes *Computer Vision—ECCV'98* (pp. 89-105): Springer.
- Tuermer, S., Kurz, F., Reinartz, P., & Stilla, U. (2013). Airborne vehicle detection in dense urban areas using HoG features and disparity maps. *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of*, 6(6), 2327-2337.
- Turer, A., Levi, A., & Aktan, A. (1998). *Instrumentation, Proof-testing and Monitoring of Three Reinforced Concrete Deck-on-steel Girder Bridges Prior to During, and After Superload*: Cincinnati Infrastructure Institute.
- Van Overschee, P., & De Moor, B. (2012). *Subspace identification for linear systems: Theory—Implementation—Applications*: Springer Science & Business Media.
- Vargas, M., Milla, J. M., Toral, S. L., & Barrero, F. (2010). An enhanced background estimation algorithm for vehicle detection in urban traffic scenes. *Vehicular Technology, IEEE Transactions on*, 59(8), 3694-3709.
- Viola, P., & Jones, M. (2001). *Rapid object detection using a boosted cascade of simple features*. Paper presented at the Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on.
- Wahba, G. (1990). *Spline models for observational data* (Vol. 59): Siam.
- Wahbeh, M. A., Caffrey, J. P., & Masri, S. F. (2003). A vision-based approach for the direct measurement of displacements in vibrating systems. *Smart Materials and Structures*.
- Wang, Y.-C., Hsieh, C.-T., Han, C.-C., & Fan, K.-C. (2014). *Vehicle Type Classification from Surveillance Videos on Urban Roads*. Paper presented at the Ubi-Media Computing and Workshops (UMEDIA), 2014 7th International Conference on.
- Webb, G., Vardanega, P., & Middleton, C. (2014). Categories of SHM Deployments: Technologies and Capabilities. *Journal of Bridge Engineering*.
- Wijnhoven, R. G., & De With, P. H. (2011). *Unsupervised sub-categorization for object detection: Finding cars from a driving vehicle*. Paper presented at the Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on.
- Wu, T., & Ranganathan, A. (2013). *Vehicle localization using road markings*. Paper presented at the Intelligent Vehicles Symposium (IV), 2013 IEEE.

- Yamaguchi, K., Kato, T., & Ninomiya, Y. (2006). *Vehicle ego-motion estimation and moving object detection using a monocular camera*. Paper presented at the Pattern Recognition, 2006. ICPR 2006. 18th International Conference on.
- Yamaguchi, T., & Hashimoto, S. (2010). Fast crack detection method for large-size concrete surface images using percolation-based image processing. *Machine Vision and Applications*, 21(5), 797-809.
- Ye, X. W., Ni, Y. Q., Wai, T. T., Wong, K. Y., Zhang, X. M., & 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. doi: 10.12989/sss.2013.12.3_4.363
- Yilmaz, A., Javed, O., & Shah, M. (2006). Object tracking: A survey. *Acm computing surveys (CSUR)*, 38(4), 13.
- Yin, Z., Wu, C., & Chen, G. (2014). Concrete crack detection through full-field displacement and curvature measurements by visual mark tracking: A proof-of-concept study. *Structural Health Monitoring*, 13(2), 205-218.
- Yu, Q., & Medioni, G. (2009). *Motion pattern interpretation and detection for tracking moving vehicles in airborne video*. Paper presented at the Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.
- Zaurin, R., & Catbas, F. (2007). Computer vision oriented framework for structural health monitoring of bridges. *IMAC XXV. Society for Experimental Mechanics*, 213-218.
- Zaurin, R., & Catbas, F. (2010). Structural health monitoring using video stream, influence lines, and statistical analysis. *Structural Health Monitoring*, 10(3), 309-332. doi: 10.1177/1475921710373290
- Zaurin, R., Khuc, T., & Catbas, F. N. (2016). Hybrid Sensor-Camera Monitoring: Case Study of a Real Bridge for Damage Detection. *Journal of Bridge Engineering*.
- Zhang, B. (2013). Reliable classification of vehicle types based on cascade classifier ensembles. *Intelligent Transportation Systems, IEEE Transactions on*, 14(1), 322-332.
- Zhang, B., Zhou, Y., & Pan, H. (2013). Vehicle classification with confidence by classified vector quantization. *Intelligent Transportation Systems Magazine, IEEE*, 5(3), 8-20.
- Zhang, Q. (2007). Statistical damage identification for bridges using ambient vibration data. *Computers & Structures*, 85(7), 476-485.
- Zhang, Z. (2000). A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(11), 1330-1334.

Zhu, Q., Xu, Y., & Xiao, X. (2014). Multiscale modeling and model updating of a cable-stayed bridge. I: Modeling and influence line analysis. *Journal of Bridge Engineering*, 04014112.

Zou, Q., Cao, Y., Li, Q., Mao, Q., & Wang, S. (2012). CrackTree: Automatic crack detection from pavement images. *Pattern Recognition Letters*, 33(3), 227-238.